Asian Journal of Information Technology 5 (9): 963-967, 2006

An Analysis of Key Management in Cryptographic Algorithms

[1]E. Ramaraj, [2]S. Karthikeyan and [3]N. Laitha
[1]Director-Computer Centre, Alagappa University, Karaikudi, Tamilnadu, India
[2]Part-Time Ph.D. Scholar, Department of Computer Science and Engineering,
Alagappa University, Karaikudi, Tamilnadu, India
[3]DJ Academy, Otthakkal Mandapam, Coimbatore-32 Tamilnadu, India

Abstract: The security of any cryptographic algorithm is based on the key size and secure key transmission. The key size depends on algorithm and firmware. The secure key transmission is very important. The aim of this study to illustrate the key distribution analysis in two scenarios. In the first scenario, to analyze how to distribute the key between sender and receiver with the presence of online Certificate Authority (CA) and also discuss some security features used in Virtual Private Network using CA. In the second scenario, to consider a fully self-organized key management for key distribution without any centralized server. Then show the comparative study in key distribution between Certificate Authority and self-organized key management.

Key words: Key distribution, cryptography, authentication, pretty good privacy, certificate authority virtual private networking

INTRODUCTION

Today, we have networked remote access systems. Machines are being linked in large numbers. They exchange large number of sensitive data. During transmission time so many hackers, used different techniques to capture the data. It is imperative for every computing professional to understand the threats and the countermeasures currently available in computing. In this age of universal electronic connectivity, of electronic eavesdropping and electronic fraud, people want to send the data in a secure way. It is not possible to send data in a private network at each an every place and cost also high[1].

We need some set of cryptographic algorithms, by help of this to transfer sensitive data easily in an insecure environment. In our view, there are two extreme ways to introduce security in a network:

- A single authority domain, where certificates and/or keys are issued by a single authority or
- A self-organization, where security does not rely on any trusted authority or fixed server[2]

We are using cryptographic algorithms either symmetric or asymmetric or both to transfer sensitive data between sender and receiver. These algorithms are not confidential and all intermediate persons know its processing functions. By using these algorithms the confidentiality of the data is based on the key values and its size. If intermediate person knows the key value he can easily trace out data and the key size is too small, using brute force attack easy to identify the data. The secret key value and its long size only protect our sensitive data to transfer in an insecure network.

In symmetric method, both the sender and receiver share the same key value. Whenever, the sender wants to send the data, he/she selects the key value and informs the receiver. The receiver uses the same key value and algorithm decrypts and identifies data. In asymmetric method or public key cryptography, the sender and receiver use different key values for encryption and decryption. The sender knows the receiver's public key. Using the receiver's public key she encrypts the data and in the receiver's place using his private key to decrypt the data. In hybrid encryption method, both symmetric and asymmetric encryption methods are integrated. In this study, the symmetric method used for encryption and asymmetric method is used for secure key transmission.

In this study we mainly discuss how to distribute the key values in secret way either using any centralized key distribution server or self-organized key management in public networks.

Key distribution server: In this study, we focus several solutions for secure key exchange between sender and receiver using centralized key distribution server.

Corresponding Author: E., Ramaraj, Director-Computer Centre, Alagappa University, Karaikudi, Tamilnadu, India

At present, more number of key exchanges is done using protocols. In this protocol, sender requires a session key for communicating to the receiver. The sender communicates to KDC, he encrypts two copies of random session key. One copy is encrypted by sender public key and another copy is encrypted by receiver public key. Both the encrypted keys send from KDC to the sender. The sender decrypts one random key using her private key and identifies the session key and sends another one copy of encrypted random session key to receiver. The receiver also decrypts random session key using the private key. This protocol relies on the absolute security of KDC.

In another method, user u receives public key of another user v from KDC. The user u randomly chooses any one-session key and encrypts with user v public key. The user v receives it and decrypted using his private key and both of the users knows session key. In this protocol, Man-in-the-Middle Attack possibility is more. Because, intermediate person w not only can he listen to messages between user u and v, he can also modify messages, delete the messages and generate totally new one. For example, whenever user u sends public key to user v, intermediate person w intercepts this key and sends to receiver v with his public key and vice versa. So, between user u and v to transfer any encrypted data, intermediate person easy to decrypt it. To avoid this attack interlock protocol has been developed by Ron Rivest and Adi Shamir.

In this protocol, u sends her public key to v and vice versa. Then, u encrypts her message using v's public key and sends half of the message to v. The user v encrypts his message using u's public key and sends half of the message to u. The user u sends other half of her encrypted message to v. User v puts the two halves of user u's message together and decrypts it with his private key and other half of his encrypted message send to user u. The user u, puts the two halves of the user v's message together and decrypts it with her private key. In this protocol, intermediate person w, substitute his public key instead of u and v. But now, when he intercepts half of user u's message, he cannot decrypt it with his private key and re-encrypt with user v public key. He must invent totally a new message and send half of it to user v.

In PGP, it uses a hybrid encryption system that combines the useful capabilities of both conventional and public key cryptography (key distribution). When the user encrypts plain text with PGP, PGP first compresses the plain text and then creates a session key, which is one-time-only secret key. This session key works with a very secure, fast conventional encryption algorithm to encrypt the plain text. Once the data is encrypted, the session key is then encrypted using recipient's public key. Every user in a public key system is vulnerable to mistaking a phony key for a real one. Validity is confidential that a public key certificate belongs to its proposed owner. In PGP, using KDC, there are three forms of trust in public key cryptography. Using Direct Trust, trust in yourself that you have correctly certified as valid the key that as an individual has told you is theirs. In Hierarchical trust, your Certificate Authority (CA) is competent and honest and is correctly certifying keys that are issued by your organization's PKI. A Web of Trust, you can download a public key from a web key server.

In, Srdjan, Jean and Levente described in Wireless Transport Layer Security (WTLS) protocol, aimed at providing secure web access from a mobile device; the servers authenticated by a certificate of their public key, delivered by a Certificate Authority (CA). This mechanism is useful, in an e-business claim in its own customers that they are connected to the right web server and that message exchange is protected. In this approach where an offline authority provides the authentication to each mobile node to join the network, it does so only the initialization of each node.

In Sardjan[3], Zhou and Haas propose a distributed public-key management service, using this private key k divided into n-independent parts and stored in n individual servers. The node receives any encrypted data from sender, first the node communicates n independent servers and to calculate private key k and using this key to decrypt the message. The major draw back in this scheme is to maintain n servers for n independent parts of a private key. The key divided into n parts based on the key size. Suppose, the key size increases servers to maintain the parts of key also increase. In Zhou[4], kong et al proposed the above solution in different way. In his study instead of n servers to maintain a key parts, doing the same job using n client machines. The major part of this scheme all the client nodes must be initialized by a trusted authority.

Virtual Private Network (VPN): VPN is the extension of a private network that encompasses links across shared or public networks like Internet. It enables you to send data between two computers across a shared or public network in a manner that emulates the properties of a point-to-point private link. There are two important protocols are used in VPN named as Point-To-Point Tunneling Protocol (PPTP) and Layer Two Tunneling Protocol (L2TP). These protocols are allowed to encrypt our data and encapsulated in the IP header. The important steps in VPN are data encryption and key management. In PPTP and L2TP using Extensible Authentication Protocol (EAP) to support variety of authentication methods, including

one-time passwords, cryptographic calculators and so on. The Microsoft Corporation implements the PPTP using Microsoft Point-To-Point Encryption (MPPE) based on RC4 algorithm from RSA[5]. MPPE for PPTP connection relies on the initial key generated during user authentication and then refreshes periodically.

Eap and certificate base authentication: In EAP, to secure integrity of the public key is the public key published with a certificate. The certificate is a data structure that is digitally signed by a CA. The certificate contains series of values such as the certificates name and usage, information for identifying the owner of the public key, expiration time and name of the CA. Remote access servers can use public key certificate for user authentication.

Self organized key management: It allows users to generate their public and private key pairs, store the values in a certificate and will be issued, to perform the user authentication without using any centralized servers. This method does not use any trusted authority. The main problem of any public key based security system is to make each users public key available to other users in such a way that the authenticity is verifiable. To avoid this problem users create the certificates. The certificate is a data structure in which a public key is bound to an identity by the digital signature of the issuer of the certificate. In this study like PGP, to generate private key and public key by users themselves. Unlike PGP, these keys are not stored in centralized servers. These certificates are stored and distributed in a fully self-organized manner.

In this method key authentication is performed via a chains of public key certificates in the following way. When a user u wants to obtain the public key of another user v, he acquires a chain of valid public-key certificates such that:

- The first certificate of the chain is directly verified by u, by using a public key that he holds and trusts.
- Each remaining certificate can be verified using the public key contained in the previous certificates of the chain.
- The last certificate contains the public key of the target user V.

To find appropriate certificate chains to other users, each ode maintains two local certificate repositories. In the non-update certificate repository of a node contains expired certificates that the node does not keep updated.

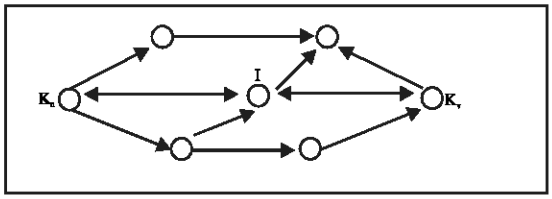

Fig. 1: Certificate graph G

The updated certificate repository of a node contains a subset of certificates that the node keeps updated. The node requests the updates for the certificates contained in its updated repository from their issuers, when or before they expire. When a user u wants to authenticate public key Kv, both networks merge their updated certificate repositories and u tries to find certificate chain to v in the merged repository. If it is found, to authenticate Kv, u tries to find certificate chains to v in its joint updated and non-updated certificate repositories.
To complete the authentication, u requests from their issuers, the updates of the expired certificates that lay on the chain. When user u wants to communicate to user v using above certificate graph the following steps are followed[2].

Step 0: User u creates own public key and private key pair.

Step 1: User u issues his public key certificates to other users based on security association about other users.

Step 2: After u identifies user v, node performs certificate exchange. The node constructs its updated certificate repository. The node can perform this operation in two ways, either by communicating with its certificate graph neighbors, For example in Fig. 1, from Ku to I and from I to Kv or by applying the repository construction algorithm. Using the above certificate graph G, the certificates on this path used by u to authenticate Kv.

Security association

Public-key based approach: It is a method to communicate between two independent users without any CA. If user u can relate the name of another user v to his public key it is called one-way security association. Both the directions two one way security association between u and v called two-way security association. A two-way security association between nodes u and v is represented by a triplet (U, Ku,au) at the side of U and a triplet (V,Kv,av) at the side of V, where U and V are the names of the users are associated with nodes u and v, Ku and Kv are public keys of u and v and au and av are the

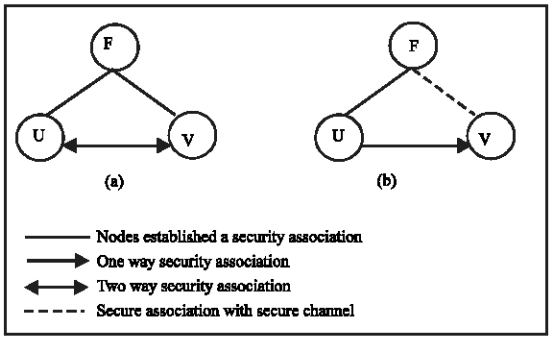

Fig. 2: Two mechanisms to create a new security association using friend

node addresses of u and v, respectively. The u and v establish short-term symmetric keys (session keys) using the public keys in the security association. To establish a process of security associations, we assume that nodes can also rely on friends[3]. In the first mechanism in Fig. 2a, two nodes u and v establish a security association if they have common friend f. Since f knows both u and v and their triplets, it can issue fresh certificates for both u and v via network. Both u and v know public key of the f and they also trust f, therefore they can both verify the received certificates and use them.

In the second study Fig. 2b, the nodes u and f are friends and f has obtained a triplet of v using secure side channel. Then, f issues a fresh certificate for the triplet of v and sends this certificate to u. Since u knows public key of f and also trusts f he can verify the received certificate and accept the received certificate if the verification is successful. Then, establish a one way security association from u to v and vice versa.

Symmetric key based approach: Using symmetric key, to reduce the computational overhead of the nodes. In this approach do not have any long-term keys. Instead, each node sets up session keys with the nodes, which it wants to communicate. In this study, u and v is represented by a triplet (U,Kuv,au) at the side of u and a triplet (V,Kuv,av) at the side of u. In this approach two way communication followed. In the first mechanism in Fig. 2a, supports the establishment of security association between two nodes u and v via a common friend f. Here, f has to establish a session key, it is generated by either by f or by u or v, in which case f would be treated as a trusted relay.

In the second mechanism in Fig. 2b, supports the establishment of security association between u and f. Here, f has to establish a session key, it is generated by either by f or by u or v, in which case f would be treated as a trusted relay and send this key between u, v and f.

COMPARITIVE STUDY

Certificate authority- key distribution:

- The distribution of public keys does not require the confidential channels
- The Certificate Authority is only to provide the key values for both sender and receiver. So, authenticity is easily verifiable.
- In this study, user does not need to be aware of the establishment of security associations.
- In the Certificate authority it is easy to organize security issues and solutions in multicast content distribution[6].
- In self-organized public key management system, each user is required to build his local certificate repository before he can use the system.
- This study is mainly useful in securing personal communications on the application level.

Self-organized key distribution:

- The distribution of public keys does not require any fixed infrastructure
- To allow users to control the security settings of the entire system
- In this study, users need to establish security association regularly
- In self-organized public key management system, each user is required to build his local certificate repository before he can use the system.
- This study is mainly useful in securing networks mechanisms such as routing in mobile ad-hoc networks.

CONCLUSION

In this study we have addressed different types of key distribution using centralized server and self-organized methods. In the first scenario, we have discussed the key distribution using centralized key distribution server in both symmetric and asymmetric encryption methods and also discuss some security features available in VPN using CA. In this study Certificate Authority performs authentication. In the second scenario, we have focussed fully on self-organized key distribution among the nodes. Certificates are stored and distributed by the nodes and each node maintains a local certificate repository and key authentication performed by via chains of certificates. The main contribution of the work can be summarized as follows:

- We discuss various methods for distributing key using centralized key distribution server and also focus its advantages and disadvantages.
- We discuss the new method for distributing the key using fully self-organized study.

REFERENCES

1. William Stallings, 2002. Cryptography and network security-principles and practice, 2nd Edn., Pearson Education Asia.
2. Srdjan Capkun, Levente Buttyan and Jean-Pierre Hubaux, 2003. Self-organized public-key management For mobile ad hoc networks, IEEE Transactions on Mobile Computing, 1: 1-13.
3. Zhou, L. and Z. Haas, 1999. Securing Ad-Hic Network, 6: 24-30.
4. Kong, J., P. Zerfos, H. Luo, S. Lu and L. Zhang, 2001. Providing robust and ubiquitous security support for mobile ad hoc networks, Proceedings of 9th Intl. Conference on Network Protocols.
5. Virtual Private, 2003. Networking with Windows Server 2003 : Overview, Microsoft Corporation.
6. Paul Judge and Mostafa Ammar, 2003. Security issues and solutions in multicast content distribution: A survey, IEEE Network, 3: 30-36.