International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

Multicasting with Key Management

Chaithra K[1], Asha N[2]

Assistant Professor, CSE, VVEIT, Mysore, India

Assistant Professor, DPGS-CEA, NIE, Mysore, India

Abstract: ***The Multicasting with Key Management allows each member to maintain a single public/secret key pair. Upon seeing the public keys of the members, a remote sender can securely broadcast to any intended subgroup chosen in an ad hoc way without depending on fully trusted authority. Even if all the non-intended members collude, they cannot extract any useful information from the transmitted messages. This system will provide an efficient member deletion/addition.***

Keywords: Cryptography, Key Management, Group Key Agreement, Broadcast Encryption, Ad-Hoc Networks

1. Introduction on Key Management

Key management is the management of cryptographic keys in a cryptosystem. This includes dealing with the generation, exchange, storage, use, and replacement of keys. It includes cryptographic protocol design, key servers, user procedures, and other relevant protocols. Key management concerns keys at the user level either between users or systems.

In the system, keys must be stored securely to maintain communications security. There are various techniques in use to do so. Likely the most common is that an encryption application manages keys for the user and depends on an access password to control use of the key.

PKI (Public Key Infrastructure): A public key infrastructure is a type of key management system that uses hierarchical digital certificates to provide authentication and public keys to provide encryption. PKIs are used in World Wide Web traffic, commonly in the form of SSL and TLS.

2. Existing Approaches for Key Management

The major security concern in group-oriented communications with access control is key management. Existing key management systems in these scenarios are mainly implemented with two approaches referred to as group key agreement (or group key exchange by some authors) and key distribution systems (or the more powerful notion of broadcast encryption).
Group key agreement allows a group of users to negotiate a common secret key via open insecure networks. Then any member can encrypt any confidential message with the shared secret key and only the group members can decrypt. In this way, a confidential intragroup broadcast channel can be established without relying on a centralized key server to generate and distribute secret keys to the potential members.

A large number of group key agreement protocols have been proposed [1], [2], [3], [4].The earlier efforts [1], [2] focused on efficient establishment of the initial group key. A tree key structure has been proposed and improved to achieve better efficiency for member joins and leaves [3], [4].

In a key distribution system, a trusted and centralized key server presets and allocates the secret keys to potential users, such that only the privileged users can read the transmitted message. The early key distribution protocol [5] does not support member addition/deletion after the system is deployed. This notion was subsequently evolved to allow the sender to freely choose the intended receiver subset of the initial group, which is usually referred to as broadcast encryption. Broadcast encryption is essential for key management [6] in priced media distribution and digital rights management.

Broadcast encryption schemes in the literature can be classified in two categories: symmetric-key broadcast encryption and public-key broadcast encryption. In the symmetric-key setting, only the trusted center generates all the secret keys and broadcasts messages to users. Hence, only the key generation center can be the broadcaster or the sender. In the public-key setting, in addition to the secret keys for each user, the trusted center also generates a public key for all the users so that any one can play the role of a broadcaster or sender. Fiat and Naor [7] first formalized broadcast encryption in the symmetric-key setting and proposed a systematic method of broadcast encryption.

In the public-key setting, Naor and Pinkas presented in [8] the first public-key broadcast encryption scheme in which up to a threshold of users can be revoked. If more than this threshold of users is revoked, the scheme will be insecure and hence not fully collusion-resistant. Subsequently, by exploiting newly developed bilinear paring technologies, a fully collusion-resistant public-key broadcast encryption scheme was presented [9] which has $O(\sqrt{N})$ complexity in key size, ciphertext size and computation cost, where N is the maximum allowable number of potential receivers.

A recent scheme [10] reduces the size of the key and the ciphertexts, although it has the same asymptotical sub-linear complexity as [9]. An up-todate scheme was presented in [11] which strengthens the security concept of public-key broadcast encryption schemes while keeping the same $O(\sqrt{N})$ complexity as [9].

3. Problem Definition

Consider a group composed of N users, indicated by $\{U_1, \ldots, U_N\}$. A sender would like to transmit secret messages to a receiver subset S of the N users, where the size of S is $n \leq$

N. The problem is how to enable the sender to efficiently and securely finish the transmission with the following constraints:

1) It is hard to deploy a key generation authority fully trusted by all users and potential senders in open network settings.
2) The communication from the receivers to the sender is limited. (e.g. in the battlefield communication setting)

According to the application scenarios, there are also some mitigating features that may be exploited for solving the problem:

1) n, the size of subgroup is usually a small or medium value, e.g. less than 256.
2) The receivers are cooperative and communicated via efficient local channels.
3) A partially trusted authority, e.g. a public key infrastructure, is available to authenticate the receivers (and the senders).

4. Advantages of Proposed System

The hybrid key management proposed formalizes the problem of secure transmission to remote cooperative groups, in which the core is to establish a one-to-many channel securely and efficiently under certain constraints.

The existing key management approaches do not provide effective solutions to this problem. On one hand, group key agreement provides an efficient solution to secure intragroup communication but, for a remote sender, it requires the sender to simultaneously stay online with the group members for multiple rounds of interactions to negotiate a common secret session key before transmitting any secret contents. This is impractical for a remote sender who may be in a different time zone. This situation is further deteriorated if the sender is mobile or otherwise dynamic.

On the other hand, broadcast encryption enables external senders to broadcast to non-cooperative members of a preset group without requiring the sender to interact with the receivers before transmitting secret contents, but it relies on a centralized key server to generate and distribute secret keys for each group member. This implies that, (i) before a confidential broadcast channel is established, numerous confidential unicast channels from the key server to each potential receiver have to be constructed, and (ii) the key server holding the secret key of each receiver can read all the communications and has to be fully trusted by any potential sender and the group members. The former requirement incurs extra costs while the latter is somewhat unrealistic in open networks.

Indeed, only very recently specific efforts were performed to secure communications from a remote sender to a cooperative group when asymmetric group key agreement was proposed by the authors at Eurocrypt 2009. In asymmetric group key agreement, the group members first negotiate a common public key but hold different secret keys. Then any sender knowing the group public key can securely encrypt to the group and only the group members can decrypt.

The instantiated protocols so far have an O(N) size public/secret key per member and does not support member deletion or addition. Subsequently, one-round asymmetric group key agreement protocols were extended to contributory broadcast encryption in which some members can be excluded but new members cannot join. The new functionality of member exclusion is at the cost of a $O(N^2)$ key size, although the cipher text size remains constant and short. The authors illustrated an efficient tradeoff with the cipher text size so that both the size of the cipher text and the size of the keys are $O(N^{2/3})$, which is still large for applications in ad hoc networks.

Second, new approach is a hybrid of group key agreement and public-key broadcast encryption. Here each group member has a public/secret key pair. By knowing the public keys of the members (e.g., by retrieving them from a public key infrastructure which is widely available in existing network security solutions), a remote sender can securely broadcast a secret session key to any intended subgroup chosen in an ad hoc way, and, simultaneously, any message can be encrypted to the intended receivers with the session key. Only the selected group members can jointly decrypt the secret session key and hence the encrypted message. In this way, the dependence on a fully trusted key server is eliminated. Also, the dynamics of the sender and the group members are coped with, because the interaction between the sender and the receivers before the transmission of messages is avoided and the communication from the group members to the remote sender is minimized. As to security, the proposal is against an attacker colluding with all the non-intended members. Even such an attacker cannot get any useful information about the messages transmitted by the remote sender

5. Components and Modules

In the Multicasting with Key Management System, each member will be maintaining a single public/secret key pair. Upon seeing the public keys of the members, a remote sender can securely broadcast to any intended subgroup chosen in an ad hoc way without depending on fully trusted authority. This system will provide an efficient member deletion/addition. The following Figure 5.1 represents the block diagram of overall System Design.

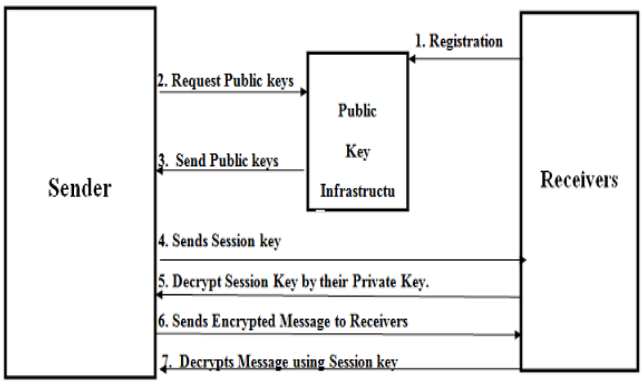

Figure 5.1: System Overall System Architecture

Multicasting with Key Management includes the following modules:

- PKI Module

- Sender Module
- Receiver Module

5.1 PKI Module

The PKI (Public Key Infrastructure) module is the one which acts as a repository of public keys of all the receiving nodes which involve in the conversation with the sender.The PKI accepts the public key of the receivers of the sub-group along with the IP address and the port number for registration. The received public key, IP addresses and port number are stored in the database.

Algorithm of PKI Module
Step 1: Create socket and wait for connection
Step 2: Accepts the request for Registration of particular receiver.
Step 3: Write the Registration information i.e., IP address, Port Number, Public key and Private key into the file.
Step 4: Send message Registration Successful.

5.2 Sender module

The Sender Module is responsible for sending the messages to the sub-group. Sender first communicates with the PKI in order to get the IP address and port number of the intended recipients and stores it into the file.

Algorithm of Sender Module
Step 1: Create socket for communication with the server.
Step 2: Gets the Public keys of the receivers to which data (text file) has to be sent.
Step 3: Gets the Session key for encrypting the actual data (text file).
Step 4: Send the Session key to the Receivers.
Step 5: Get the data (text file) which has to be sent to receivers
Step 6: Encrypt the data using the Session key.
Step 7: Send the Encrypted data (text file)

5.3 Receiver Module

Receiver module is responsible for creating key pair i.e., Private –Public keys for each of the receiver node in the sub-group after it registers into the PKI by sending IP address, port number and public keys of all intended receivers of the group. It receives the session key from the sender and decrypts the session key. When it receivers the actual message from the sender, it decrypts the message by using the Session key.

Algorithm of Receiver Module
Step 1: Create Socket and establish connection with Sender.
Step 2: If the receivers' IP address is registered in the PKI server then Perform Step 3
Step 3: Store the public keys in an array
Step 4: Compare the session key with the public keys of the receivers stored in the array in Step 3. If the comparison is successful then go to Step 5 else go to Step 6.
Step 5: Decrypt the Session Key.
Step 6: Display the Session key in encrypted form.
Step 7: If the decrypted session key is obtained then go to Step 8. If not got to Step 9.
Step 8: Decrypt the file using the Session key and display the contents.
Step 9: File contents are not displayed

6. Conclusion and Future Enhancement

6.1 Conclusion

The idea behind the hybrid approach of traditional broadcast encryption and group key agreement is to overcome the obstacles of the potentially limited communication from the group to the sender, the unavailability of a fully trusted key generation center and the dynamics of the sender.

- In this system, each member will be maintaining a single public/secret key pair.
- Upon seeing the public keys of the members, a remote sender can securely broadcast to any intended subgroup chosen in an ad hoc way without depending on fully trusted authority.
- Even if all the non-intended members collude, they cannot extract any useful information from the transmitted messages.
- This system will provide an efficient member deletion/addition.

6.2 Future Enhancement

Currently this project is implemented using Java and Eclipse over the LAN.

- It can be enhanced to work on the Mobile Ad-hoc Networks.
- Also it can be implemented Using NS-2 and can be used to measure the system performance.
- Instead of Encrypting Session Key using AES, Diffie-Helman Key Exchange process can be incorporated

References

[1] M. Burmester and Y. Desmedt, "A Secure and Efficient ConferenceKey Distribution System," in Advances in Cryptology–EUROCRYPT'94,LNCS, vol. 950, pp. 275-286, 1995.
[2] M. Waldvogel, G. Caronni, D. Sun, N. Weiler and B. Plattner, "TheVersaKey Framework: Versatile Group Key Management," IEEE J. Sel.Areas Commun., vol. 17, no. 9, pp. 1614-1631, Sept. 1999.
[3] Sherman and D. McGrew, "Key Establishment in Large Dynamic Groups Using One-way Function Trees," IEEE Trans. Software Eng., vol. 29, no. 5, pp. 444-458, May 2003.
[4] Y. Kim, A. Perrig and G. Tsudik, "Tree-Based Group Key Agreement," ACM Trans. Inf. Syst. Security, vol. 7, no. 1, pp. 60-96, Feb. 2004.
[5] Ingemarsson, D.T. Tang and C.K. Wong, "A Conference on Key Distribution System," IEEE Trans. Inf. Theory, vol. 28, no. 5, pp. 714-720, Sep. 1982.
[6] M. Abdalla, Y. Shavitt and A. Wool, "Key Management for Restricted Multicast Using Broadcast Encryption," IEEE/ACM Trans. Netw., vol.8, no. 4, pp. 443-454, Aug. 2000.

[7] Fiat and M. Naor, "Broadcast Encryption," in Advances in Cryptology–CRYPTO'93, LNCS, vol. 773, pp. 480-491, 1993.
[8] M. Naor and B. Pinkas, "Efficient Trace and Revoke Schemes," in Proc.4th International Conf. on Financial Cryptography (FC'00), LNCS, vol.1962, pp. 1-20, 2001.
[9] D. Boneh, C. Gentry and B. Waters, "Collusion Resistant Broadcast Encryption with Short Ciphertexts and Private Keys," in Advances in Cryptology–CRYPTO'05, LNCS, vol. 3621, pp. 258-275, 2005.
[10] J.-H. Park, H.-J. Kim, M.-H. Sung and D.-H. Lee, "Public Key Broadcast Encryption Schemes With Shorter Transmissions," IEEE Trans. On broadcasting, vol. 54, no. 3, pp. 401-411, Sep. 2008.
[11] C. Gentry and B. Waters, "Adaptive Security in Broadcast Encryption Systems (with Short Ciphertexts)," Advances in Cryptology–EUROCRYPT'09, LNCS, vol. 5479, pp. 171-188, Springer-Verlag, 2009.

Author Profile

Chaithra K has received B.E in ISE in SJCE, Mysore and pursuing M.Tech in CNE, NIE, Mysore. She is working as Assistant Professor in the Department of Computer Science and Engineering in Vidya Vikas Institute of Engineering and Technology, Mysore.

Asha N has completed her Post Graduation in Computer Network Engineering from VTU. Her research interests are in mobile ad hoc networks and cloud computing. She is working as Assistant Professor in the Department of PG Studies, NIE, Mysore.