International Journal of Current Advanced Research
ISSN: O: 2319-6475, ISSN: P: 2319 – 6505, Impact Factor: SJIF: 5.995
Available Online at www.journalijcar.org
Volume 6; Issue 7; July 2017; Page No. 4874-4876
DOI: http://dx.doi.org/10.24327/ijcar.2017.4876.0603

Research Article

THIRD-PARTY AUTHENTICATION KEY PROTOCOLS

Satheesh Kumar.N[1] and Satyanarayana N*[2]

[1]Department of CSE, Rayalaseema University, Kurnool

[2]Department of CSE, Nagole Institute of Technology & Sciences Hyderabad.A.P. INDIA

A R T I C L E I N F O

Article History:

Received 27th April, 2017
Received in revised form 10th May, 2017
Accepted 6th June, 2017
Published online 28th July, 2017

Key words:

MANET, UCB, Protocol

A B S T R A C T

This work presents key sharing protocols to safeguard security in large networks, using in new directions in classical cryptography. Two third-party key sharing protocols, one with implicit user authentication and the other with explicit mutual authentication, are proposed to demonstrate the merits of the new combination, which include the following.
First, Security against such attacks as man-in-the-middle, eavesdropping and replay. Second, Efficiency is improved as the proposed protocols contain the fewest number of communication rounds among existing key sharing protocols. Third, Two parties can share and use a long-term secret (repeatedly). To prove the security of the proposed schemes, this work also presents a new primitive called the Unbiased-Chosen Basis (UCB) assumption. The securities we are trying to provide are

1. Username/Password authentication for admin
2. Per-user, feature-by-feature, and field-by-field access control
3. Granular administrative privileges

INTRODUCTION

In order to transfer data from one system to another we need to have network between systems. During transmission there is chance of our valuable data to be hacked by unauthenticated users. In order to avoid, cryptography has came in to existence. Classical cryptography cannot detect the existence of passive attacks such as eavesdropping.

Network eavesdropping or network sniffing is a network layer attack consisting of capturing packets from the network transmitted by others computers and reading data content in search of sensitive information like passwords, session tokens or any kind of confidential information Key sharing protocols are used to facilitate sharing secret session keys between users on communication networks. By using these shared session keys, secure communication is possible on insecure public networks. However, various security problems exist in poorly designed key sharing protocols; for example, a malicious attacker may derive the session key from the key distribution process. A legitimate participant cannot ensure that the received session key is correct or fresh and a legitimate participant cannot confirm the identity of the other participant. Designing secure key distribution protocols in communication security is a top priority. In some key distribution protocols, two users obtain a shared session key via a trusted center (TC).

Corresponding author:* **Satheesh Kumar.N
Department Of CSE, Rayalaseema University, Kurnool

Since three parties (two users and one TC) are involved in session key negotiations, these protocols are called three-party key distribution protocols, as in contrast with two-party protocols here only the sender and receiver are involved in session key negotiations.

The main objective of this paper is to verify the session key from trusted center and sender which improve key verification and secure the communication and also to identify the security threads in session key verification.

Existing System

In classical cryptography, three-party key distribution protocols utilize challenge response mechanisms or timestamps to prevent replay attacks. However, challenge response mechanisms require at least two communication rounds between the TC and participants, and the timestamp approach needs the assumption of clock synchronization which is not practical in distributed systems (due to the unpredictable nature of network delays and potential hostile attacks). Furthermore, classical cryptography cannot detect the existence of passive attacks such as eavesdropping. This fact can then be used to reduce the number of rounds of other protocols based on challenge-response mechanisms to a trusted center (and not only three-party authenticated key distribution protocols).

Limitations

Disadvantage of separate process 3AQKDP and 3AQKDPMA were provide the authentication only for message, to identify

the security threads in the message. Not identify the security threads in the session key.

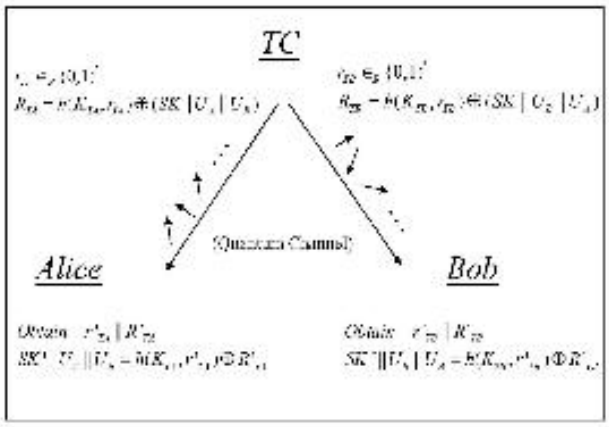
Proposed System

In cryptography, key sharing protocols employ mechanisms to distribute session keys and public discussions to check for eavesdroppers and verify the correctness of a session key. However, public discussions require additional communication rounds between a sender and receiver and cost precious qubits. By contrast, classical cryptography provides convenient techniques that enable efficient key verification and user authentication.

There are two types of Key sharing Protocols, they are

Aksp

This section describes the details of the 3AKSP by using the notations defined in previous sections. Here, we assume that every participant shares a secret key with the TC in advance either by direct contact or by other ways.



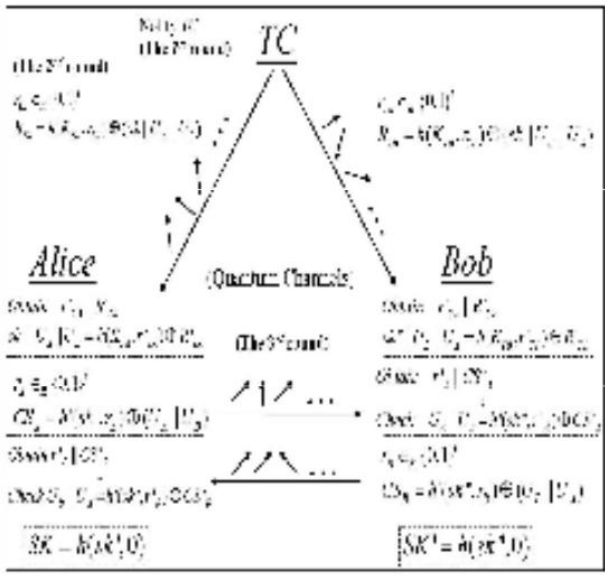
KSPMA

The proposed 3KSPMA can be divided into two phases: the Setup Phase and the Key sharing Phase. In the Setup Phase, users preshare secret keys with the TC and agree to select polarization bases of qubits based on the preshared secret key. The Key Sharing Phase describes how Alice and Bob could share the session key with the assistance of TC and achieve the explicit user authentication.

Implementation

Cryptography easily resists replay and passive attacks, whereas classical cryptography enables efficient key verification and user authentication. By integrating the advantages of both classical cryptography, this work presents two KSPs with the following contributions:

- Man-in-the-middle attacks can be prevented, eavesdropping can be detected, and replay attacks can be avoided easily
- User authentication and session key verification can be accomplished in one step without public discussions between a sender and receiver
- The secret key preshared by a TC and a user can be long term (repeatedly used); and
- The proposed schemes are first provably secure QKDPs under the random oracle model.



In the proposed KSPs, the TC and a participant synchronize their polarization bases according to a preshared secret key. During the session key distribution, the preshared secret key together with a random string are used to produce another key encryption key to encipher the session key. A recipient will not receive the same polarization qubits even if an identical session key is retransmitted.

Consequently, the secrecy of the preshared secret key can be preserved and, thus, this preshared secret key can be long term and repeatedly used between the TC and participant. Due to the combined use of classical cryptographic techniques with the quantum channel, a recipient can authenticate user identity, verify the correctness and freshness of the session key, and detect the presence of eavesdroppers. Accordingly, the proposed KSPs require the fewest communication rounds among existing KSPs. The same idea can be extended to the design of other KSPs with or without a TC. The random oracle model is employed to show the security of the proposed protocols. The theory behind the random oracle model proof indicates that when the adversary breaks the three-party KSPs, then a simulator can utilize the event to break the underlying atomic primitives. Therefore, when the underlying primitives are secure, then the proposed three-party KSPs are also secure.

Key management is the provisions made in a cryptography system design that are related to generation, exchange, storage, safeguarding, use, vetting, 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. This is in contrast to key scheduling; key scheduling typically refers to the internal handling of key material within the operation of a cipher.

Successful key management is critical to the security of a cryptosystem. In practice it is arguably the most difficult aspect of cryptography because it involves system policy, user training, organizational and departmental interactions, and coordination between all of these elements.

These concerns are not limited to cryptographic engineering. Key management requires both technical and organizational

decisions, and as a result, some aspects of key management risk being neglected by managers and engineers, out of concern that the problem is technical or managerial, respectively.

5. CONCLUSIONS

This study proposed two third-party KSPs to demonstrate the advantages of combining classical cryptography with new technology. Compared with classical third-party key sharing protocols, the proposed KSPs easily resist replay and passive attacks. Compared with other KSPs, the proposed schemes efficiently achieve key verification and user authentication and preserve a long term secret key between the TC and each user. Additionally, the proposed KSPs have fewer communication. Rounds than other protocols. Although the requirement of the quantum channel can be costly in practice, it may not be costly in the future. Moreover, the proposed KSPs have been shown secure under the random oracle model. By combining the advantages of classical cryptography with quantum cryptography, this work presents a new direction in designing KSPs.

The Proposed system is an efficient, authenticated, scalable key agreement for large and dynamic multicast systems, which is based on the bilinear map. Compared with the Existing system, we use an identity tree to achieve the authentication of the group member.

Bibliography

1. G. Li, "Efficient Network Authentication Protocols: Lower Bounds and Optimal Implementations," *Distributed Computing*, vol. 9, no. 3, pp. 131-145, 1995.
2. Kehne, J. Schonwalder, and H. Langendorfer, "A Nonce-Based Protocol for Multiple Authentications," *ACM Operating Systems Rev.*, vol. 26, no. 4, pp. 84-89, 1992.
3. M. Bellare and P. Rogaway, "Provably Secure Session Key Distribution: The Three Party Case," Proc. 27th ACM Symp. *Theory of Computing*, pp. 57-66, 1995.
4. J. Nam, S. Cho, S. Kim, and D. Won, "Simple and Efficient Group Key Agreement Based on Factoring," Proc. Int'l Conf. *Computational Science and Its Applications* (ICCSA '04), pp. 645-654, 2004.
5. H.A. Wen, T.F. Lee, and T. Hwang, "A Provably Secure Three-Party Password-Based Authenticated Key Exchange Protocol Using Weil Pairing," *IEE Proc. Comm.*, vol. 152, no. 2, pp. 138-143, 2005.
6. J.T. Kohl, "The Evolution of the Kerberos Authentication Service," *EurOpen Conf. Proc.*, pp. 295-313, 1991.
7. B. Neuman and T. Ts'o, "Kerberos: An Authentication Service for Computer Networks," *IEEE Comm.*, vol. 32, no. 9, pp. 33-38, 1994.
8. W. Stallings, Cryptography and Network Security: Principles and Practice 3/e. Prentice Hall, 2003.
9. K.-Y. Lam and D. Gollmann, "Freshness Assurance of Authentication Protocols," Proc. European Symp. Research in Computer Security (ESORICS '92), pp. 261-271, 1992.
10. R. Shirey, Internet Security Glossary, IETF RFC 2828, May 2000

How to cite this article:

Satheesh Kumar.N and Satyanarayana N (2017) 'Third-Party Authentication Key Protocols ', *International Journal of Current Advanced Research,* 06(07), pp. 4874-4876. DOI: http://dx.doi.org/10.24327/ijcar.2017.4876.0603

****