

Key Establishment for General Ad Hoc Network

Zhan Liu and Mi Lu

Department of Electrical and Computer Engineering

Texas A & M University

College Station, Texas 77840, U.S.A.

{liuzhan, mlu}@ee.tamu.edu

Tel: 979 845 9578

Fax: 979 845 2630

Keywords: key establishment, general ad hoc network, hash function

which can use the same key without using timestamp.

Abstract

In this paper, we present a general ad hoc network and the session secret conception. By using session secret, key cache and reuse without using timestamp is possible. A combination of protocols is presented, which is suitable for general ad hoc network and any two participants who communicate with each other very often.

Notation used

A and B :	The two principals
S :	A server trusted by A and B
K_{AB} :	Session key between A and B
K_{AB}^0, K_{AB}^i :	Session key
$\{N_a\}_{K_{AB}}$:	Encrypt N_a with key K_{AB}
$[N_a]_{K_{AB}}$:	hash of N_a with key K_{AB}
$h(M)$:	hash message M
N_A, N_B, K_S :	Nonce generated by A, B, S
T_A :	Timestamp
L :	Lifetime of a key

1 Introduction

For ad hoc network security, we assume that a location-limited channel [4]-[5] exists. However, a general assumption is that participants have access to a server at very beginning. Later, they lost the server and form an ad hoc network. For example, members of a rescue team have access to a server at home. They went to the field and lose the access to the server or any server. Therefore, they form an ad hoc network. This kind of ad hoc network is what we called the general ad hoc network. This paper presents a key establishment protocol for this kind of network.

For two principals who communicate with each other very often, there is the need to cache the same key and reuse it for many times until the key is expired. However, the cache and reuse key usually need timestamp, like the Kerberos [1] protocol. In this paper, we present a combination of protocols,

2 Kerberos

We use Kerberos as an example for key cache and reuse.

The Kerberos [3] is as following:

1. $A \rightarrow S : A, B, N_A$
2. $S \rightarrow A : \{K_{AB}, B, L, N_A, \dots\}_{K_{AS}}, \{K_{AB}, A, L, \dots\}_{K_{AS}}$
3. $A \rightarrow B : \{A, T_A\}_{K_{AB}}, \{K_{AB}, A, L, \dots\}_{K_{AS}}$

The session key K_{AB} is cached and reused before it expires.

3 Combination of Protocols

Instead of using K_{AB} directly as the session key, we use it as a shared-secret between A and B . We called it session secret. The session key is derived from the

session secret as $K_{AB}^0 = h(K_{AB})$, which is calculated by A and B separately.

Since server-based Boyd protocol [3] provides key authentication, key freshness and key confirmation, we based our protocol on server-based Boyd protocol.

3.1 Boyd protocol

1. $A \rightarrow S : A, B, N_A$
2. $S \rightarrow B : \{A, B, K_S\}K_{AS}, \{A, B, K_S\}K_{BS}, N_A$
3. $B \rightarrow A : \{A, B, K_S\}K_{AS}, [N_A]K_{AB}, N_B$
4. $A \rightarrow B : [N_B]K_{AB}$

Where $K_{AB} = MAC_{K_S}(N_A, N_B)$

3.2 Our Combination Protocol

Instead of using K_{AB} as session key, we keep it as a shared-secret and calculate the session key as $K_{AB}^0 = h(K_{AB})$ and use K_{AB}^0 as the session key. The protocol is as following:

1. $A \rightarrow S : A, B, N_A$
2. $S \rightarrow B : \{A, B, K_S\}K_{AS}, \{A, B, K_S\}K_{BS}, N_A$
3. $B \rightarrow A : \{A, B, K_S\}K_{AS}, [N_A]K_{AB}^0, N_B$
4. $A \rightarrow B : [N_B]K_{AB}^0$

Since A and B now shared a secret, next time, when they need to communicate with each other, they can establish a new session key by using server-less Boyd protocol [3] as following:

1. $A \rightarrow B : N_A$
2. $B \rightarrow A : N_B$

The new session key will be calculated by A and B as $K_{AB}^i = h(K_{AB}, N_A, N_B)$

For two principals who communicate often, this protocol reduce the communication between A , B and the server S .

Besides, one kind of application of this protocol is that A and B lost connection with the server after key establishment. Then A and B form an ad hoc network. For example, A and B are kind of members of rescue team. A and B use the server to share a secret at home before they depart to their destination. So they lost the connection to the server. They form an ad hoc network.

4 Conclusion

By keeping the session key as a session secret and deriving a new session key from the session secret, we have a combination protocol for a general ad hoc network. We also show that ad hoc network always need two steps for key establishment and general way of key establishment for ad hoc network.

With this combination protocol, we can also reuse the session secret many times without using timestamp. This also reduces the communication between A , B and S .

Besides, using server-less Boyd protocol instead of server-based Boyd protocol, greatly reduce the transmission size for both A and B . Therefore, reduce energy consumption for A and B , which is crucial for battery-powered devices [7].

By hashing, we can blind a shared secret and reuse it for many times.

References

- [1] B. Clifford Neuman et al, "Kerberos: An authentication service for computer networks", *IEEE Communications Magazine*, 32(9):33-38, September 1994
- [2] Colin Boyd, "A class of flexible and efficient key management protocols", *8th IEEE Computer Security Foundations Workshop*, pp. 2-8, 1996
- [3] Colin Boyd, et al, "Protocols for Authentication and Key Establishment", *Springer*, 2003
- [4] Dirk Balfanz, et al, "Talking to Strangers: Authentication in Ad-Hoc Wireless Networks"
- [5] Frank Stajano, et al, "The resurrection duckling - what next?", *Proc. Eighth Security Protocols workshop, Lecture Notes in Computer Science 2133, Springer-Verlag, Berlin, 2001*, pp. 204-214
- [6] A. Menezes, et al, "Handbook of Applied Cryptography", *CRC Press, Inc. 1997*, Chapter 1
- [7] J.D. Tygar, et al, "SPINS:Security protocols for sensor networks", *Wireless Networks*, v8, n5, Sept. 2002, pp521-534