

# Key Establishment Protocol for Computation-limited Devices

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, forward secrecy, key compromise impersonation

## Abstract

*This paper present a new protocol based on symmetric encryption, which has quasi-forward secrecy and resistance to quasi-key compromise impersonation. It is a protocol suitable for computation-limited devices. Also by blinding the long-term key, we make the attack to long-term key harder. We have done the theoretical analysis and will finish the simulation and implementation.*

## 1 Introduction

Forward secrecy and resistance to key compromise impersonation [6] are two important properties of security protocol. However, both of them require asymmetric encryption. Forward secrecy requires that when a long-term key compromised, session keys that were previously established using that long-term key should not compromise. Key compromise impersonation means that an intruder can use the compromised long-term key to impersonate the principal, who legally has the long-term key. A good protocol should provide both forward secrecy and resistance to key compromise impersonation. However, both forward secrecy and resistance to key compromise impersonation need asymmetric encryption. Asymmetric encryption usually needs more computational power than symmetric encryption and hash function. As the development of tech-

nology, many devices, such as DVD players, MP3 players and TV sets will have Internet access ability. However, those devices are all computation-limited devices, which generally don't have the capability to do asymmetric encryption. A heterogeneous network with many computation-limited devices, such as MP3 players, will be the trend of future. One of those heterogeneous networks is home network [20]. A secure protocol for heterogeneous network based on symmetric encryption is needed. In this paper, quasi-forward secrecy and quasi-key compromise impersonation conceptions will be proposed and a new protocol, which provides quasi-forward secrecy and quasi-key compromise impersonation, is proposed.

### Notation used

---

A and B: The two principals  
S: A server trusted by A and B  
I: An intruder  
 $I_A$ : An intruder pretends to be A  
 $I_B$ : An intruder pretends to be B  
 $K_{AB}$ : Session key between A and B  
 $K_{AS}, K_{BS}$ : Long-term key between A, B and S  
 $K_{AS}^i, K_{BS}^i$ : Long-term session key for A, B  
 $E_T(M)$ : Encryption of message M using the public key  $K_T$   
 $Sig_A(M)$ : Signature with appendix of message M by principal A  
 $\{N_a\}K_{AB}$ : Encrypt  $N_a$  with key  $K_{AB}$   
 $[N_a]K_{AB}$ : One-way transformation of  $N_a$  with key  $K_{AB}$   
 $h(M)$ : hash message M  
 $N_A, N_B, K_S$ : Nonce generated by A, B, S

---

## 2 Forward Secrecy and Key Compromise Impersonation

**Definition 1** *A key establishment protocol provides forward secrecy if compromise of the long-term keys of a set of principals does not compromise the session keys establishment in previous protocol runs involving those principals.*

**Definition 2** *A protocol provides resistance to key compromise impersonation if compromise of the long-term keys of a principal A does not allow the intruder to impersonate A for future communication.*

### 2.1 Protocols Provided Forward Secrecy

General speaking, there are two kinds of protocols [6], which provide forward secrecy. One is key transport protocol and another is key establishment protocol. For comparison, they are rewritten as following.

#### 2.1.1 Key Transport Protocols Provided Forward Secrecy

1.  $A \rightarrow B : K_T, N_A, \text{Sig}_A(K_T, B)$
2.  $B \rightarrow A : E_T(K_{AB}, \text{Sig}_B(h(K_{AB}), A, N_A))$

$K_{AB}$  is the session key. After the key transportation, the ephemeral public/private key pair  $E_T$  are destroyed.

#### 2.1.2 Key Establishment Protocols Provided Forward Secrecy

1.  $A \rightarrow S : A, B$
2.  $A \rightarrow B : A, g^{N_A}$
3.  $S \rightarrow B : \{A, B, K_S\}K_{BS}$
4.  $S \rightarrow A : \{A, B, K_S\}K_{AS}$
5.  $B \rightarrow A : B, g^{N_B}$

A calculate the session key  $K_{AB} = (g^{N_B})^{N_A K_S}$  and B calculate the session key  $K_{AB} = (g^{N_A})^{N_B K_S}$

From the above two protocols, it is easy to see that asymmetric encryption capability is required for A and B.

It is quite obvious that there is no resistance to key compromise impersonation for the two protocols. An intruder who compromise long-term key  $K_{AS}$  has the ability to impersonate A. Therefore, he can communicate with anyone whom A can communicate with in the name of A and get all the sensitive documents which are for A only.

### 2.2 Protocol Provided Resistance to Key Compromise Impersonation

MTI protocols [6] provide resistance to key compromise impersonation. The protocols are all based on Diffie-Hellman algorithm, therefore require asymmetric encryption capability.

## 3 Quasi-forward Secrecy and Quasi-key Compromise Impersonation

Since many devices in a heterogeneous network, such as a home network, don't have the asymmetric encryption capability, forward secrecy and resistance to key compromise impersonation are not possible. However, a quasi-forward secrecy and resistance to quasi-key compromise impersonation (we will define them later) are possible by using hash function. Our protocol, which provided quasi-forward secrecy and resistance to quasi-key compromise impersonation, is based on Boyd protocol [6]. We choose Boyd protocol, since Boyd protocol provides key authentication, key freshness and key confirmation.

### 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.  $A \rightarrow B : [N_B]K_{AB}$

The session key is calculated by A and B independently as

$$K_{AB} = \text{MAC}_{K_S}(N_A, N_B)$$

It is quite obvious that Boyd protocol has no forward secrecy. If  $K_{AS}$  compromise, an intruder can have  $K_S$  by decrypting  $\{A, B, K_S\}K_{AS}$ . Since  $N_A$

and  $N_B$  are transmitted in plaintext, the intruder now can calculate

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

Therefore, any previous and future communication with A using long-term key  $K_{AS}$  will compromise.

The protocol has no resistance to key compromise impersonation. By compromise the long-term key  $K_{AS}$ , an intruder can impersonate A as following:

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

The similar thing will happen if the long-term key of B  $K_{BS}$  compromise. This time, all the previous and future communications with B using long-term key  $K_{BS}$  will compromise and the intruder can impersonate B.

## 3.2 Our Protocol Provided Quasi-forward Secrecy and Resistance to Quasi-key Compromise Impersonation

### 3.2.1 Protocol One

Our protocol is as following:

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

Where  $K_{AS}^i = h(K_{AS}, N_A)$ ,  $K_{BS}^i = h(K_{BS}, N_A)$

We call  $K_{AS}$  and  $K_{BS}$  long-term key and  $K_{AS}^i$  and  $K_{BS}^i$  long-term session key.

Since  $K_{AS}(K_{BS})$  is only a shared-secret between A(B) and S. They are never used directly to encrypt any message. So an intruder can do ciphertext-only attack [18] only against  $K_{AS}^i$  and  $K_{BS}^i$ . And they only have limited amount of ciphertext for each long-term session key. Hash function is a one-way function, getting  $K_{AS}^i$  and  $K_{BS}^i$  will not in any way help to get  $K_{AS}$  and  $K_{BS}$ .

Of course, compromise  $K_{AS}$  will still compromise all the previous and future communication with principal A. However, compromise  $K_{AS}^i$  will compromise only the corresponding session, not any previous communication with A. The reason is that the long-term session key is different for different round of key establishment.

But compromise  $K_{AS}^i$  still give the an active intruder the chance to impersonate A. The intruder can replay an old message he just compromised as following:

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

Since the random number  $N_A$  is old, the long-term session key  $K_{AS}^i = h(K_{AS}, N_A)$ , which compromise, is old too. Even though  $K_S$  is new, the intruder can get it by decrypt  $\{A, B, K_S\}K_{AS}^i$ . Now he has enough information to calculate the session key  $K_{AB} = MAC_{K_S}(N_A, N_B)$ . This means that the intruder can impersonate as A to communicate with anyone A can communicate with.

If the long-term session key  $K_{BS}^i$  compromise, the intruder can impersonate B as following:

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

The intruder impersonate as B and start a new session by using an old  $N_A$  he eavesdropped. Since  $N_A$  is old, the long term-session key  $K_{AS}^i = h(K_{AS}, N_A)$  and  $K_{BS}^i = h(K_{BS}, N_A)$  are old too. The intruder can decrypt  $\{B, A, K_S\}K_{BS}^i$  and get  $K_S$ . Then he can calculate the session key  $K_{AB} = MAC_{K_S}(N_A, N_B)$  and finish the protocol. Actually, if the long-term session key  $K_{BS}$  compromise, the intruder can impersonate B and communicate with anyone B can communicate with and get whatever B can get.

### 3.2.2 Protocol Two

To avoid this attack, the protocol should like the following:

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

Now the long-term session keys are calculated as  $K_{AS}^i = h(K_{AS}, N_A, N_S)$ ,  $K_{BS}^i = h(K_{BS}, N_A, N_S)$

This time, no matter how many  $K_{AS}^i$  compromise, only the corresponding sessions compromise. An intruder can not impersonate A or B as in protocol one. Let's say if he try the attack as before:

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

Even though he still can use an old  $N_A$ , the long-term session key  $K_{AS}^i = h(K_{AS}, N_A, N_S)$  and  $K_{BS}^i = h(K_{BS}, N_A, N_S)$  are new because  $N_S$  is new. The intruder can not calculate new  $K_{AS}^i$  and  $K_{BS}^i$  because he has no knowledge of  $K_{AS}$  and  $K_{BS}$ . The intruder can not impersonate A or B.

However, if an intruder can impersonate as both A and S, he still can apply the following attack.

1. omit
2.  $I_S \rightarrow B : \{A, B, K_S\}K_{AS}^i, \{A, B, K_S\}K_{BS}^i, N_A, N_S$
3.  $B \rightarrow I_A : \{A, B, K_S\}K_{AS}^i, [N_A]K_{AB}, N_B, N_S$
4.  $I_A \rightarrow B : [N_B]K_{AB}$

The intruder can impersonate as S and send B an old message he eavesdropped. Since  $\{A, B, K_S\}K_{AS}^i$  is old and compromise. The intruder already knew  $K_S$ . Because  $N_A, N_S$  are old, B will calculate  $K_{BS}^i = h(K_{BS}, N_A, N_S)$ , which is old too. So by decrypting  $\{A, B, K_S\}K_{BS}^i$ , B will get an old  $K_S$  and calculate the session key  $K_{AB} = MAC_{K_S}(N_A, N_B)$ , which is old too. There B and the intruder agree with an old session key which is compromised. Therefore, we lose the resistance to quasi-key compromise impersonation.

### 3.2.3 Protocol Three

The following protocol will avoid the attack.

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

In step 2, B can calculate the long-term session key  $K_{BS}^i = h(K_{BS}, N_A, N_B)$ . In step 3, S can calculate the long-term session key  $K_{BS}^i = h(K_{BS}, N_A, N_B)$  and  $K_{AS}^i = h(K_{AS}, N_A, N_B)$ . In step 4, A can calculate the long-term session key  $K_{AS}^i = h(K_{AS}, N_A, N_B)$ . And finally A and B agree a session key  $K_{AB} = MAC_{K_S}(N_A, N_B)$ .

It is quite obvious that we have quasi-forward secrecy.

A can be sure that  $K_{AS}^i$  is fresh because  $N_A$  is fresh. A also be sure that nobody can forge it because an intruder does not know long-term key  $K_{AS}$ , therefore, A is sure that  $\{A, B, K_S\}K_{AS}^i$  is fresh. So A is sure that  $K_S$  is fresh. And finally he is sure session key  $K_{AB}$  is fresh too. B can be sure for the freshness of session key  $K_{AB}$  in the same way.

Therefore, this protocol give us both quasi-forward secrecy and resistance to quasi-key compromise impersonation.

**Definition 3** A key establishment protocol provides quasi-forward secrecy if compromise of the long-term keys of a set of principals does not compromise the session keys establishment in previous protocol runs involving those principals. However, compromise of the long-term session key of a set of principals does not compromise the session keys establishment in previous protocol runs involving those principals.

**Definition 4** A protocol provides resistance to quasi-key compromise impersonation if compromise of the long-term keys of a principal A does not allow the intruder to impersonate A for future communication, but compromise long-term session key does not.

So protocol one and two provide only quasi-forward secrecy and protocol three provides both quasi-forward secrecy and resistance to quasi-key compromise impersonation.

Since compromise the long-term session key will only compromise the corresponding session, there is no need to make the long-term session key more secure than the session key. The key size of the long-term session key could be comparable to the session key.

## 4 Conclusion

We have presented a new server-based key establishment protocol based on symmetric encryption with both quasi-forward secrecy and resistance to quasi-key compromise impersonation. We achieved these properties by introducing long-term session key and hiding the long-term key. The long-term key is protected against ciphertext-only attack because of the using of long-term session key. This also makes the long-term key more secure against attack. Therefore, his lifetime can be longer.

Our protocol makes the communication between computation-limited devices, which can only do symmetric encryption, more secure.

## References

- [1] Amtana Dutta-Roy, "Network for homes", *IEEE spectrum*, Dec. 1999
- [2] Anand R. Prasad, et al, "Security architecture for wireless LANs: corporate & public environment", *IEEE VTC 2000*, pp.283 - 287
- [3] N. Asokan, et al, "Key Agreement in Ad-hoc Network", 2000
- [4] Brent Miller, "Home networking device and service discovery requirements", <http://www.watersprings.org/pub/id/draft-miller-homedisc-req-00.text>
- [5] Brian Tung, "Kerberos A Network Authentication System", Addison-Wesley, 1999
- [6] Colin Boyd, et al, "Protocols for Authentication and Key Establishment", *Springer*, 2003
- [7] Dirk Balfanz, et al, "Talking to Strangers: Authentication in Ad-Hoc Wireless Networks"
- [8] Dritan Kaleshi, et al, "Ensuring interoperability in a home networking system: a case study", *IEEE Trans. on Consumer Electronics*, vol.45, No.4, Nov. 1999, pp 1134-1143
- [9] Frank Stajano, et al, "The resurrection duckling: security issues for ubiquitous computing", *Security & Privacy-2002*, pp22-26
- [10] Frank Stajano, et al, "The resurrecting duckling: security issues in ad-hoc wireless networks", *Proc. Seventh security Protocols Workshop, lecture Notes in Computer Science 1796, Springer-Verlag, Berlin 2000*, pp. 172-182
- [11] 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
- [12] Frazer Bennett et al, "Piconet: Embedded mobile networking", *IEEE Personal Communications*, Oct. 1997, pp. 8 - 15
- [13] Harney, et al, "Group Key Management Protocol Architecture", *Request for comments (RFC) 2093*, Internet Engineering Task Force, March 1997
- [14] Hideaki Nakakita, et al, "A study on secure wireless networks consisting of home appliances", *IEEE International Conference on Consumer Electronics*, 2002, 2002 Digest of Technical Papers, Jun 18-20 2002, Atlanta, GA, p 178-179
- [15] Jae-Chul Moon, et al, "Multi agent architecture for intelligent home network service using tuple space model", *IEEE Trans. on consumer Electronics*, v 46, n3, Aug, 2000, p 791 - 794
- [16] Jae-Min Lee, et a, "A new home network protocol for controlling and monitoring home appliances-HNCP", *Digest of Technical Papers - IEEE International conference on Consumer Electronics*, 2002, pp. 312 - 313
- [17] J.M. Kahn, et al, "Next century challenges: mobile networking for "smart dust"", *Proc. MOBI-COM, 1999*, pp271 - 278
- [18] A. Menezes, et al, "Handbook of Applied Cryptography", *CRC Press, Inc. 1997*, Chapter 1

- [19] Peter M. Corcoran, et al, "Wireless home network infrastructure for wearable appliances", *IEEE international Conference on Consumer Electronics*, 2002, 104-105
- [20] Prashant Krishnamurthy, et al, "Security in wireless residential networks," *IEEE transactions on consumer electronics*, v 48, n 1, Feb., 2002, p 157 - 166
- [21] Rolf Oppliger, "Authentication systems for secure networks", *Artech House*, 1996
- [22] Rudolf Volner, "Home Security System and CATV", *IEEE Annual International Carnahan Conference on Security Technology, Proceedings, 2001*, pp 293-301
- [23] Sarvar Patel, "Number theoretic attacks on secure password schemes", *IEEE Symposium on Security and Privacy*, pp. 236 -247, 1997
- [24] Shujun Li, et al, "Chaotic encryption scheme for real-time digital video", *SPIE*, 2002.
- [25] Stephen Thomas, "SSL and TLS essentials", *John Wiley & Sons, Inc*, 2000
- [26] Steven M. Bellovin, et al, "Encrypted key exchange: Password-based protocols secure against dictionary attacks", *IEEE Symposium on Research in Security and Privacy*, pp. 72-84, 1992
- [27] Steven M. Bellovin, et al, " Augmented encrypted key exchange: A password-based protocol secure against dictionary attacks and password file compromise", *1st ACM Conference on Computer and Communications Security*, pp.244-250, 1993
- [28] Steve Burnett et al, " RSA Security's official Guide to Cryptography", *RSA Press*, 2001
- [29] Tai-Yeon Ku et al, "A java-based architecture supporting IEEE 1394 for home entertainment network ", *IEEE international Conference on Consumer Electronics*, 2002, 104-105
- [30] E. Topalis, et al, "Generic network management architecture targeted to support home automation networks and home Internet connectivity", *Digest of Technical Papers - IEEE International conference on Consumer Electronics*, 1999, pp. 42 - 43
- [31] J.D. Tygar, et al, "SPINS:Security protocols for sensor networks", *Wireless Networks*, v8, n5, Sept. 2002, pp521-534
- [32] T. Zahariadis, et al, "Multimedia home networks standards and interfaces", *Computer Standards and Interfaces*, v24, n5, Nov. 2002, pp 425 - 435