Modification The Hsu-Harn-Mu-Zhang-Zu Group Key Establishment Protocol

Caesario Oktanto Kisty
Badan Siber dan Sandi Negara
Jakarta, Indonesia
caesario.oktanto@bssn.go.id

Sofu Risqi Yulian Saputra
Badan Siber dan Sandi Negara
Jakarta, Indonesia
sofu.risqi@bssn.go.id

***Abstract*—Security in wireless network is an important aspect that must be considered. Because every data that is exchanged can be accessed by anyone. In case, protocol for key establishment. It requires certain methods to provide security feature. One of many protocol for key establishment is proposed by Hsu et al. They proposed a protocol provides much lower computation complexity and storage complexity. They claimed that the group key is key freshness, key confidentiality, and key authentication. But, in another paper there are flawed for its claimed. The protocol is not key authentication. In this paper we try to modified the protocol. The root cause of issue is protocol can not be assured that the key is right from KGC. Then, we modified protocol by adding digital signature scheme. So, the new protocol can be fullfil key authentication feature.**

Keywords—Key Establishment Protocol, Wireless Network, Digital Signature, Public Key Infrastructure.

I. INTRODUCTION

Hsu et al [1] proposed a new protocol (HHMZZ protocol) to provide authenticated group key establishment in a wireless network. Hsu et al state that HHMZZ protocol achieves the security claims with key freshness, key confidentiality, and key authentication. In proof of Theorem 1, state that any insider cannot forge a group key without being detected since the group key is a function of each member's longterm secret x_i. Shortly afterward Mitchell J [2] describe a serious security issue with that scheme. He claimed that the proof is incorrect. The insider can forge a group key in HHMZZ protocol. Based on this insecure, we modified HHMZZ protocol so it can meet key authentication security feature.

The rest of this paper is organized as follows. Next section describe the protocol include security claim. Section 3, we describe vulnerability in the protocol by [2]. Section 4, we describe modification of HHMZZ protocol and analysis. The paper concludes in section 4.

II. THE HSU-HARN-MU-ZHANG-ZU PROTOCOL

A. Overview

There are two types of key establishment protocol, key transfer protocol and key agreement protocol. HHMZZ is a key transfer protocol for group communication. The transfer's key is a group key, or any subset (group) of community -under trusted Key Generation Centre (KGC), as a trusted third party- has a same key for communication. HHMZZ uses combination of cryptogaphic hash function and a linear secret sharing scheme based on Vandermonde matrix. This scheme allows lower computation and storage complexity, but at least has the same security degree.

B. The Protocol

The protocol consists of pre-distributing phase and group key distributing phase.

1) Pre-distributing phase

There are some following reqiurements for use the protocol which set up in this phase.

First, KGC choose a finite field $K = Z_p$, where p is a safe large prime;

Second, protocol must agree on two cryptographic hash-functions h_1 and h_2, both mapping to K;

Third, each user i registers at KGC and choose a secret key $x_i \in K$ which is shared with the KGC in a secure manner;

Fourth, protocol must agree on the function $\boldsymbol{v}_m(x) = (1, x, x^2, \dots, x^m)$ where $\boldsymbol{v}_m: K \to K^{m+1}$ and $m \geq 2$.

2) Group key distributing phase

Generally, KGC will receive a group key generation request before randomly select a group key. Then KGC will distribute this group key to all group registered member in a secure and authenticated manner.

Assume that there are t members in a group $\{1, \dots, t\}$ and corresponding with shared secrets $x_1, x_2, \dots, x_t$. The protocol proceeds as follows:

Step 1, the initiator request group key generation to the KGC with list of group member $\{1, \dots, t\}$.

Step 2, as a response, KGC broadcast the list of all participating member which is requested.

Step 3, each participating group member select a random challange $r_i \in K$ and send to KGC.

Step 4, KGC randomly selects a group key $S \in K$ and $r_0 \in K$. KGC compute the inner product $\left(\boldsymbol{v}\left(x_i \oplus h_1(x_i, r_i, r_0)\right), \boldsymbol{r}\right) = s_i$ and $u_i = (S - s_i) mod\ p$ for each participating group member, in which vector $\boldsymbol{r} = (r_0, r_1, \dots, r_t)$. KGC compute $Auth = h_2(S, 1, \dots, t, r_0, r_1, \dots, r_t, u_1, \dots, u_t)$ and broadcast $\{Auth, r_0, u_i\}$ to all group member, $i = 1, \dots, t$.

Step 5, for each group member computes $\left(\boldsymbol{v}\left(x_i \oplus h_1(x_i, r_i, r_0)\right), \boldsymbol{r}\right) = s_i$ using secret key x_i. Then, each group member compute the group key $S = (u_i + s_i) mod\ p$ using public information u_i from KGC. Finally, each goup member verifies the $Auth$ by recomputing using computed group key.

978-1-5386-9559-3/18/$31.00 ©2018 IEEE

C. Security Claim

The proposed protocol has the following security claim:

1. This protocol achieves security features with key freshness, key confidentiality, and key authentication.
2. This protocol can resist attacks in both synchronous and asynchronous networks.
3. This protocol achieves backward and forward secrecy of group communication.
4. This protocol can resist attack in both outside and inside attacker.

III. Protocol Insecure Claim

Mitchell [2] claim that proof of theorem 1 is incorrect. Consequence is an insider can forge a group key. Insider can be a man-in-the-middle between KGC and victim. Insider replace the original group key by his malicious group key which only use hash function as a authentication. Mitchell [2] describe a serious vulnerability in following analysis:

The attack occur when KGC broadcast $\{Auth, r_0, u_i\}$ to all group member. For example, let U as a set of member which requested by one of them (as a initiator) to achieved a group key from KGC. The scenario is that victim user or U_v and malicious user or U_m is a group member of U, hence $U_v, U_m \in U$.

Attack scenario, U_m is replaced broadcast value from KGC to U_v by his a new group key self-generated. As a valid member of U, U_m can calculate the group key S which generated and distributed by KGC. By chooses a different group key S^* and u_v is a public parameter, he compute,

$$u_v^* = u_v - S + S^*$$

and

$$Auth^* = h_2(S^*, 1, \dots, t, r_0, r_1, \dots, r_t, u_1, \dots, u_v^*, \dots u_t)$$

Then U_m send a modified broadcast value from KGC, where $Auth$ and u_v are replaced by $Auth^*$ and u_v^*. Its is simple for U_v to verify correctly a malicious group key from U_m.

IV. Modification Proposed Protocol

A. Attack Analysis

The fundamental issue in the protocol is that the security features claimed by the author are not fully key authentication. Principally Hsu et al only ensure that the value being broadcast by KGC is still intact. Although victim can verify the value obtained from KGC, the generated group key can not be ascertained, that the group key is correct generated by KGC.

The impact of such vulnerabity is the insider can replace brodcasted value from KGC. But the other members can not be sure that the value u_i obtained is correct from KGC. Because when insider is known the group key and using the formula (to compute u_v^* above), insider can eliminate s_i although insider not known. On other hand, it is a consequence of using Vandermonde matrix. Therefore, there is a need for a method so that each user can verify that the value obtained is correctly from KGC. One method that can be used is digital signature.

B. Digital Signature

Digital signature is an authentication mechanism that enables the creator of a message to attach a code that acts as a signature [3]. The code is obtained from encrypting the hash of message using the creator's private key. Each other user which has the creator's public key can verify the authentication of message. Verifier can ensure that the receive message is correct created by the creator. Moreover, if the message is verified signed by the creator, he can not repudiate that the message is not created by him. Illustration of digital signature scheme, as shown in Fig. 1. and Fig. 2.

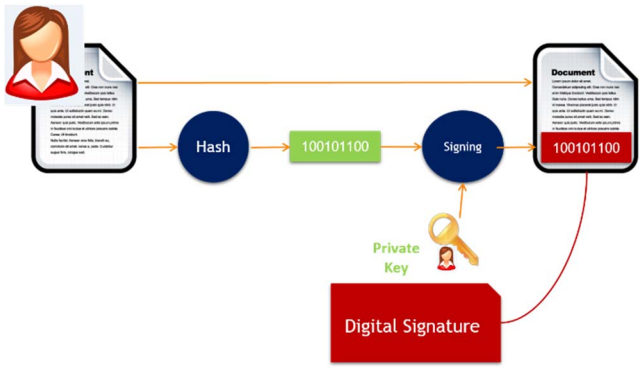

Fig.1. Signing Process

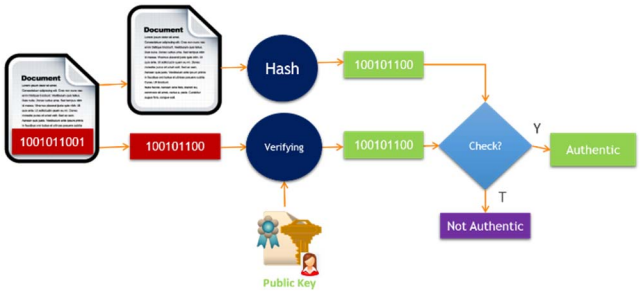

Fig.2. Verification Process

Digital signature is based on public key cryptosystem. Mostly digital signature algorithms are Digital Signature Algorithm (DSA), RSA Digital Signature Algorithm, Elliptic Curve Digital Signature Algorithm, etc. Such algorithm state before is digital signature standard [4] issued by National Institute of Standards and Technology (NIST). However, as the development of quantum computer technology issues, the above algorithm is vulnerable for later use [5]. Thus, it is necessary to develop a post quantum cryptography algorithm as the solution of the such issue [6]. But, this paper does not address the issue.

C. Modification HHMZZ Protocol

Based on the vulnerability that exists in the proposed protocol. The following section will explain the modifications HHMZZ protocol.

The protocol consists of pre-distributing phase and group key distributing phase.

1) Pre-distributing phase

Generally, this phase similar with original protocol. Beside set up existing requirement, KGC is set up public key pair (Public and Private key). KGC keeps the private key in a secure manner. Meanwhile the public key is shared to each registered user.

2) Group key distributing phase

Similar with original protocol, KGC will receive a group key generation request before randomly select a group key. The protocol proceeds as shown in Fig. 3.

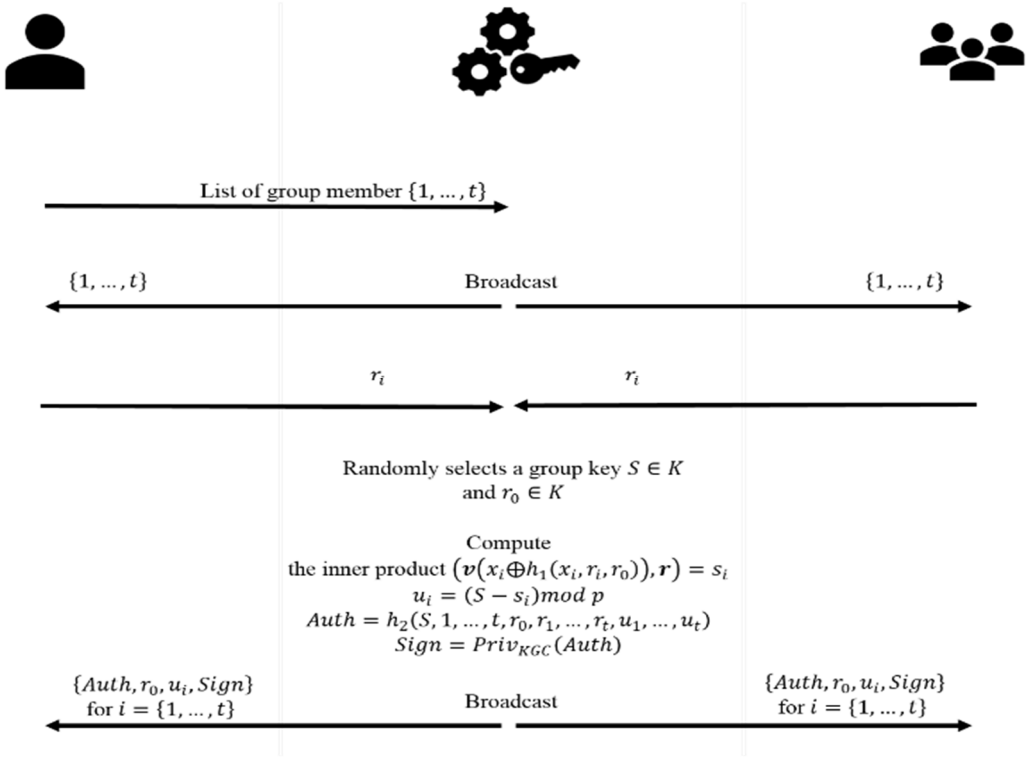

Fig. 3 Modification HHMZZ Protocol

After receive broadcasted value from KGC, then they compute:

$$\left(\boldsymbol{v}\left(x_i \oplus h_1(x_i, r_i, r_0)\right), \boldsymbol{r}\right) = s_i \text{ using secret key } x_i$$
$$S = (u_i + s_i) mod\ p$$

For ensure the key authentication which is obtained, each user verify signing message using KGC's public key, compute as follow:

$$Auth = h_2(S, 1, \dots, t, r_0, r_1, \dots, r_t, u_1, \dots, u_t)$$
$$Ver = Pub_{KGC}(Sign)$$

Check $Auth =? Ver$, if the result is same then user convinced that the group key is generated by KGC.

D. Security Analysis

Security advantages of this protocos is almost similar with security claim in Section II. Security claim and proof is similar too with [1], except in theorem 1 about key authentication. Specificly, security in key authentication features is different. Since the modification is applied, insider cannot forge group key as explained by [2], or in other words forged group key can be detected. This following forgery group key attack resistant scenario for modified protocol:

1. U_m is malicious in the group;
2. When KGC broadcast last parameter, U_m chooses a different group key S^*, and compute:

$$u_v^* = u_v - S + S^*$$

and

$$Auth^* = h_2(S^*, 1, \dots, t, r_0, r_1, \dots, r_t, u_1, \dots, u_v^*, \dots u_t)$$

3. U_m send a modified broadcast value from KGC to victim or U_v, where $Auth$ and u_v are replaced by $Auth^*$ and u_v^*.
4. U_v using Vandermonde matrix function to extract the group key. Cause modified value by U_m, U_v get a group key S^*. Then U_v check the value is which received, compute as follow:

$$Auth^* = h_2(S^*, 1, \dots, t, r_0, r_1, \dots, r_t, u_1, \dots, u_v^*, \dots u_t)$$
$$Ver = Pub_{KGC}(Sign)$$

5. Check $Auth^* =? Ver$, and the result is different. U_v can be ensured that the group key S^* is not generated by KGC and has been modified in the middle. So, he can notified to KGC that the group key is compromised.

CONCLUSION

As explanation of modification protocol above, we claimed that the modification is fullfil key authentication criteria. Each user in group can ensure that the group key is generated by KGC. Compromised group key can be detected while user check digital signature of KGC.

REFERENCES

[1] C. F. Hsu, L. Harn, Y. Mu, M. Zhang, and X. Zhu. "Computation-efficient key establishment in wireless group communications". Wireless Networks, 23:289-297, 2017.

[2] C. J. Mitchell. "The Hsu-Harn-Mu-Zhang-Zu group key establishment protocl is insecure". arXiv:1803.05365 [cs.CR], March 2018, 6 pages.

[3] B. Colin. "Information Security and Cryptography – Protocols for Authentication and Key Establishment". London: Springer, c2003 xxiv, 321 p. : ill. ; 24 cm.

[4] Federal Information Processing Standards Publication, Gaithersburg. FIPS PUB 186-4, "Digital Signature Standard (DSS)". July 2013

[5] C. Lily et al. "Report on Post-Quantum Cryprography". National Institute of Standards and Technology – Internal Report 8105, http://dx.doi.org/10.6028/NIST.IR.8105 .

[6] M. Dustin, F. Larry, W. Greg. "Securing Tomorrow's Information Through Post-Quantum Cryptography". Information Technology Laboratory (ITL) Bulletin fo February 2018.