Ephemeral key compromise attack on the IB-KA protocol

Qingfeng Cheng, Chuangui Ma

Zhengzhou Information Science and Technology Institute
Zhengzhou 450002, P. R. China
qingfengc2008@sina.com

Abstract. Recently, Dario Fiore and Rosario Gennaro proposed the IB-KA protocol, which was inspired by MQV protocol. They provide a full proof of security of IB-KA protocol using techniques developed by Krawczyk in the Canetti-Krawczyk model. They designed the IB-KA protocol with some security properties such as perfect forward secrecy, reflection attack resilience, and key compromise impersonation resilience. But they didn't consider ephemeral key compromise problem in the design of IB-KA protocol, and made no analysis whether the IB-KA protocol can resist ephemeral key compromise attacks. In this paper, we present ephemeral key compromise attack on the the IB-KA protocol. Our work shows that the IB-KA protocol is designed without ephemeral key compromise resilience.

Key words: Ephemeral key compromise attack; Authenticated key exchange; Canetti-Krawczyk model.

1 Introduction

Ephemeral key compromise (EKC) resilience was first proposed by Krawczyk [1] in 2005. Krawczyk point out that Many applications may boost protocol performance by pre-computing ephemeral pair (x, g^x) for later use in the protocol, and these stored pairs are more vulnerable to leakage than static private key. In 2007, LaMacchia, Lauter and Mityagin [2] added Ephemeral key reveal query in the extended Canetti-Krawczyk (eCK) model, which was based on the Canetti-Krawczyk model [3]. Their motivation to include revelations of ephemeral keys in the model comes from practical (i.e. engineering) considerations and scenarios such as active adversarial attacks or compromise of the random number generator (RNG) used by one of the parties. In the eCK model, the adversary can make reveal queries on the test session according to the freshness definition, and reveal both ephemeral private keys, both static keys, or one of each from the two different parties. Based on these reasons, it seems to be reasonable to put more emphasis on the EKC resilience.

In recent, Fiore D and Gennaro R proposed a new identity based key agreement protocol, called IB-KA protocol [4], which was inspired by MQV protocol [5, 6]. They provided a full proof of security in the Canetti-Krawczyk model [3].

But they didn't consider ephemeral key compromise problem in the design of IB-KA protocol, and made no analysis whether the IB-KA protocol can resist ephemeral key compromise attacks. In this paper, we will show that their protocol cannot resist ephemeral key compromise attack.

The rest of this paper is organized as follows. In Section 2 we introduce preliminaries used in this paper. In Section 3 we review the IB-KA protocol. In section 4 we present an attack on the IB-KA protocol. In the final section, we conclude this paper.

2 Preliminaries

Let p and q be primes, where $q|p-1$. Let $G =< g >$ be a multiplicative subgroup of Z_p^*, of prime order q.

- **Computational Diffie-Hellman (CDH) Problem:** Given $U = g^u, V = g^v \in G$, where u, v were drawn at random from Z_q, compute $W = g^w \in G$, such that $CDH(U, V) = W$. That is, compute $g^w = g^{uv} \bmod p$.
- **Decisional Diffie-Hellman (DDH) Problem:** Given $U = g^u, V = g^v, W = g^w \in G$, where u, v, w were drawn at random from Z_q, determine whether $DDH(U, V, W) = 1$ or not. That is, determine whether $w = uv \bmod q$ or not.
- **Gap Diffie-Hellman (GDH) Problem:** Given $U = g^u, V = g^v \in G$, where u, v were drawn at random from Z_q, as well as an oracle that solves the DDH problem on G, compute $g^w = g^{uv} \bmod p$.

We say that G satisfies the GDH assumption if no feasible adversary can solve the GDH problem with non-negligible probability.

3 Review of the IB-KA Protocol

In this section, we briefly review the IB-KA protocol. For more details about the IB-KA protocol, refer to [4].

Protocol setup.

The Key Generation Center (KGC) chooses a group G of prime order q, a random generator $g \in G$ and two hash functions $H_1 : \{0,1\}^* \rightarrow Z_q$ and $H_2 : Z_q \times Z_q \rightarrow \{0,1\}^l$. Then it picks a random $x \in Z_q$ and sets $y = g^x$. Finally the KGC outputs the public parameters (G, g, y, H_1, H_2) and keeps the master secret key x for itself.

Key Derivation.

A user with identity ID receives, as its secret key, a Schnorr's signature of the message $m = ID$ under public key y. More specifically, the KGC after verifying the user's identity, creates the associated secret key as follows. First it picks a random $k \in Z_q$ and sets $r_{ID} = g^k$. Then it uses the master secret key x to compute $s_{ID} = k + H_1(ID; r_{ID})x$. (r_{ID}, s_{ID}) is the secret key returned to the user. The user can verify the correctness of its secret key by using the public key y and checking the equation $g^{s_{ID}} = r_{ID} y^{H_1(ID, r_{ID})}$.

In the IB-KA protocol, two parties A and B perform the following steps.

Step1: Party A selects $t_A \in Z_q$ randomly, computes $u_A = g^{t_A}$, then sends (A, r_A, u_A) to party B.

Step2: Upon receiving the message (A, r_A, u_A), party B selects $t_B \in Z_q$ randomly, computes $u_B = g^{t_B}$, and sends (B, r_B, u_B) to party A. Then party B computes the session key $Z = H_2(z_1, z_2)$, where $z_1 = (u_A r_A y^{H_1(A, r_A)})^{t_B + s_B}$ and $z_2 = u_A^{t_B}$.

Step3: Upon receiving the message (B, r_B, u_B), party A computes the session key $Z = H_2(z_1, z_2)$, where $z_1 = (u_B r_B y^{H_1(B, r_B)})^{t_A + s_A}$ and $z_2 = u_B^{t_A}$.

4 Analysis of the IB-KA Protocol

In this section, we present EKC attack on the IB-KA protocol. In fact, if the adversary can learn the ephemeral private key t_A or t_B, he will mount the attack successfully and generate the same session key as party A or party B. The adversary E can carry out his attack as follows:

Step1: Party A selects $t_A \in Z_q$ randomly, computes $u_A = g^{t_A}$, then sends (A, r_A, u_A) to party B.

Step2: The adversary E intercepts the message (A, r_A, u_A). Then E chooses $t_E \in Z_q$ and computes $u_B^* = \frac{g^{t_E}}{g^{s_B}}$. Finally, he impersonates party B to send the message (B, r_B, u_B^*) to party A.

Step3: Upon receiving the message (B, r_B, u_B^*), party A computes the session key $Z = H_2(z_1, z_2)$, where $z_1 = (u_B^* r_B y^{H_1(B, r_B)})^{t_A + s_A}$ and $z_2 = (u_B^*)^{t_A}$.

Since we assume the adversary can get t_A, the adversary can learn $-t_A$ easily. Then he can compute z_1, z_2 as

$$g^{t_E t_A}(g^{s_A})^{t_E} == g^{t_E t_A} g^{t_E s_A} = (g^{t_E})^{t_A + s_A} = (\frac{g^{t_E}}{g^{s_B}} g^{s_B})^{t_A + s_A} = (u_B^* g^{s_B})^{t_A + s_A} = (u_B^* r_B y^{H_1(B, r_B)})^{t_A + s_A} = z_1$$

$$g^{t_E t_A}(g^{s_B})^{-t_A} = g^{t_E t_A - s_B t_A} = (g^{t_E - s_B})^{t_A} = (\frac{g^{t_E}}{g^{s_B}})^{t_A} = (u_B^*)^{t_A} = z_2$$

It means that the adversary E can generate the same session key as party A. So we have successfully launched this attack on the IB-KA protocol. The attack to party B can mount similarly.

5 Conclusion

Since EKC problem was pointed out by Krawczyk in 2005, there are many protocols designed with EKC resilience. However, there are still some protocols without ECK resilience. In this paper, we analyze the IB-KA protocol and show that the IB-KA protocol proved secure in the Canetti-Krawczyk using Krawczyk's method cannot resist EKC attack.

References

1. Krawczyk, H.: HMQV: A High-Performance Secure Diffie-Hellman Protocol, In: CRYPTO 2005. Lecture Notes in Computer Science 3621, 2005, pp. 546-566. Full version available at http://eprint.iacr.org/2005/176.
2. LaMacchia, B., Lauter,K., Mityagin, A.: Stronger Security of Authenticated Key Exchange, In: ProvSec 2007, Lecture Notes in Computer Science 4784, 2007, pp.1-16.
3. Canetti, R., Krawczyk, H.: Analysis of Key-Exchange Protocols and Their Use for Building Secure Channels, In: EUROCRYPT 2001, Lecture Notes in Computer Science 2045, 2001, pp. 453-474.
4. Fiore, D., Gennaro, R.: Making the Diffie-Hellman Protocol Identity-Based. Available at http://eprint.iacr.org/2009/174.
5. Menezes A., Qu M., and Vanstone S.: Some new key agreement protocols providing mutual implicit authentication, Second Workshop on Selected Areas in Cryptography (SAC 95), 1995.
6. Law L., Menezes A., Qu M., Solinas J., and Vanstone S.: An efficient Protocol for Authenticated Key Agreement, Designs, Codes and Cryptography, vol.28,pp.119-134, 2003.