Cryptanalysis of a New Efficient Authenticated Multiple-Key Exchange Protocol from Bilinear Pairings

Qingfeng Cheng

Department of Language Engineering, Luoyang University of Foreign Languages
Luoyang 471003, P.R. China
(Email: qingfengc2008@sina.com)

(Received Dec. 11, 2012; revised and accepted Mar. 13, 2013)

Abstract

Multiple-key exchange (MKE) protocols allow two parties to generate two or more shared session keys over insecure networks. In recent years, many MKE protocols have been proposed. However, most of them still have some security flaws. In this letter, we will analyze a new MKE protocol proposed by Farash et al. in 2012, and present two attacks against Farash et al.'s protocol.

Keywords: Basic impersonation attack, key compromise impersonation attack, multiple-key exchange, parallel session attack

1 Introduction

Multiple-key exchange (MKE) is a process in which two parties can calculate two or more session keys in one session. The pioneer work in the field was proposed by Harn and Lin[4] in 1998. Since then many MKE protocols have been proposed by researchers, such as [2, 5-11]. However, most of them have been proven insecure due to lack of some desirable security attributes.

Recently, Farash et al.[3] showed that Cheng and Ma's protocol[1] was insecure against a forgery attack. Further, they proposed an enhanced MKE protocol based on bilinear pairings, called FAAJ protocol. They also claimed that their protocol satisfied all known security requirements. In this letter, we will show that the FAAJ protocol cannot resist the basic impersonation attack. In addition, we also prove that the FAAJ protocol is insecure against the combination of key compromise impersonation attack and parallel session attack.

2 Review of FAAJ Protocol

2.1 System Initialization Stage

Let k be a security parameter, q be a large prime, G_1 be an additive group of prime order q and G_2 be a multiplicative group of the same prime order q. P is a generator of group G_1. $e: G_1 \times G_1 \to G_2$ is a bilinear pairing. The system's public parameters are $\{q, G_1, G_2, P, e\}$. Each party U chooses $x_U \in Z_q^*$ randomly as a private key and computes $Y_U = x_U P$ as a public key. In addition, each party U has a certificate $Cert(Y_U)$ issued by a certificate authority (CA). For more details about the FAAJ protocol, refer to [3].

2.2 Key Agreement Stage

In the following description we suppose party A and party B wish to communicate with each other.

Step 1: Party A randomly chooses $a_1, a_2 \in Z_q^*$, and computes $T_{A1} = a_1 P, T_{A2} = a_2 P$. If $k_{A1} \cdot k_{A2} \neq 0$, then A computes

$$S_A = (k_{A1} \cdot k_{A2}) x_A - (a_1 k_{A1} + a_2 k_{A2}) \bmod q .$$

Otherwise, A chooses a new random number and start the session again. Finally, A sends $\{T_{A1}, T_{A2}, S_A, Cert(Y_A)\}$ to party B.

Step 2: Upon receiving $\{T_{A1}, T_{A2}, S_A, Cert(Y_A)\}$, B verifies $Cert(Y_A)$. If it fails, B terminates the session. Otherwise, B checks the following equation

$$(k_{A1} \cdot k_{A2}) Y_A = S_A P + (k_{A1} T_{A1} + k_{A2} T_{A2}) .$$

If the above equation fails, B terminates the session. Otherwise, B randomly chooses $b_1, b_2 \in Z_q^*$, and computes $T_{B1} = b_1 P, T_{B2} = b_2 P$. If $k_{B1} \cdot k_{B2} \neq 0$, then B computes

$$S_B = (k_{B1} \cdot k_{B2}) x_B - (b_1 k_{B1} + b_2 k_{B2}) \bmod q .$$

Otherwise, B chooses a new random number and computes T_{B1}, T_{B2} again. Finally, B sends $\{T_{B1}, T_{B2}, S_B, Cert(Y_B)\}$ to party A and generates the session keys as follows:

$$K_1 = e(T_{A1}, T_{A1})^{b_1 b_1} = e(P,P)^{a_1 a_1 b_1 b_1}$$
$$K_2 = e(T_{A1}, T_{A1})^{b_1 b_2} = e(P,P)^{a_1 a_1 b_1 b_2}$$
$$K_3 = e(T_{A1}, T_{A1})^{b_2 b_2} = e(P,P)^{a_1 a_1 b_2 b_2}$$
$$K_4 = e(T_{A1}, T_{A2})^{b_1 b_1} = e(P,P)^{a_1 a_2 b_1 b_1}$$
$$K_5 = e(T_{A1}, T_{A2})^{b_1 b_2} = e(P,P)^{a_1 a_2 b_1 b_2}$$
$$K_6 = e(T_{A1}, T_{A2})^{b_2 b_2} = e(P,P)^{a_1 a_2 b_2 b_2}$$
$$K_7 = e(T_{A2}, T_{A2})^{b_1 b_1} = e(P,P)^{a_2 a_2 b_1 b_1}$$
$$K_8 = e(T_{A2}, T_{A2})^{b_1 b_2} = e(P,P)^{a_2 a_2 b_1 b_2}$$
$$K_9 = e(T_{A2}, T_{A2})^{b_2 b_2} = e(P,P)^{a_2 a_2 b_2 b_2}$$

Step 3: Upon receiving $\{T_{B1}, T_{B2}, S_B, Cert(Y_B)\}$, A verifies $Cert(Y_B)$. If it fails, A terminates the session. Otherwise, A checks the following equation

$$(k_{B1} \cdot k_{B2}) Y_B = S_B P + (k_{B1} T_{B1} + k_{B2} T_{B2}).$$

If the above equation fails, A terminates the session. Otherwise, A generates the session keys as follows:

$$K_1 = e(T_{B1}, T_{B1})^{a_1 a_1} = e(P,P)^{a_1 a_1 b_1 b_1}$$
$$K_2 = e(T_{B1}, T_{B2})^{a_1 a_1} = e(P,P)^{a_1 a_1 b_1 b_2}$$
$$K_3 = e(T_{B2}, T_{B2})^{a_1 a_1} = e(P,P)^{a_1 a_1 b_2 b_2}$$
$$K_4 = e(T_{B1}, T_{B1})^{a_1 a_2} = e(P,P)^{a_1 a_2 b_1 b_1}$$
$$K_5 = e(T_{B1}, T_{B2})^{a_1 a_2} = e(P,P)^{a_1 a_2 b_1 b_2}$$
$$K_6 = e(T_{B2}, T_{B2})^{a_1 a_2} = e(P,P)^{a_1 a_2 b_2 b_2}$$
$$K_7 = e(T_{B1}, T_{B1})^{a_2 a_2} = e(P,P)^{a_2 a_2 b_1 b_1}$$
$$K_8 = e(T_{B1}, T_{B2})^{a_2 a_2} = e(P,P)^{a_2 a_2 b_1 b_2}$$
$$K_9 = e(T_{B2}, T_{B2})^{a_2 a_2} = e(P,P)^{a_2 a_2 b_2 b_2}$$

3 Analysis of FAAJ Protocol

3.1 Attack 1

In this subsection, we present the first attack against the FAAJ protocol. We will show that the FAAJ protocol cannot resist the basic impersonation attack.

The adversary E can mount the basic impersonation attack as follows:

Step 1: Party A randomly chooses $a_1, a_2 \in Z_q^*$, and computes $T_{A1} = a_1 P, T_{A2} = a_2 P$. If $k_{A1} \cdot k_{A2} \neq 0$, then A computes

$$S_A = (k_{A1} \cdot k_{A2}) x_A - (a_1 k_{A1} + a_2 k_{A2}) \bmod q.$$

Otherwise, A chooses a new random number and start the protocol again. Finally, A sends $\{T_{A1}, T_{A2}, S_A, Cert(Y_A)\}$ to party B.

Step 2: Upon intercepting $\{T_{A1}, T_{A2}, S_A, Cert(Y_A)\}$, the adversary E randomly chooses $c \in Z_q^*$ and computes $T_{B1}^* = k_{B2}^* Y_B, T_{B2}^* = cP$ and $S_B^* = -ck_{B2}^*$. Then the adversary E impersonates party B and sends $\{T_{B1}^*, T_{B2}^*, S_B^*, Cert(Y_B)\}$ to party A.

Step 3: Upon receiving $\{T_{B1}^*, T_{B2}^*, S_B^*, Cert(Y_B)\}$, A verifies $Cert(Y_B)$. If it fails, A terminates the session. Otherwise, A checks the following equation

$$(k_{B1} \cdot k_{B2}) Y_B = S_B^* P + (k_{B1} T_{B1}^* + k_{B2} T_{B2}^*),$$

where k_{B1} is the x-coordinate value of T_{B1}^* and k_{B2} is the x-coordinate value of T_{B2}^*, clearly $k_{B2} = k_{B2}^*$.

Since

$$\begin{aligned} S_B^* P + (k_{B1} T_{B1}^* + k_{B2} T_{B2}^*) &= -ck_{B2}^* P + (k_{B1} T_{B1}^* + k_{B2} T_{B2}^*) \\ &= -ck_{B2}^* P + (k_{B1} k_{B2}^* Y_B + k_{B2} cP) \\ &= -ck_{B2}^* P + (k_{B1} k_{B2} Y_B + k_{B2}^* cP) \\ &= -ck_{B2}^* P + k_{B1} k_{B2} Y_B + ck_{B2}^* P \\ &= k_{B1} k_{B2} Y_B, \end{aligned}$$

the verification holds. Then A will generate the session keys as follows:

$$K_1 = e(T_{B1}^*, T_{B1}^*)^{a_1 a_1}$$
$$K_2 = e(T_{B1}^*, T_{B2}^*)^{a_1 a_1}$$
$$K_3 = e(T_{B2}^*, T_{B2}^*)^{a_1 a_1} = e(cP, cP)^{a_1 a_1}$$
$$K_4 = e(T_{B1}^*, T_{B1}^*)^{a_1 a_2}$$
$$K_5 = e(T_{B1}^*, T_{B2}^*)^{a_1 a_2}$$
$$K_6 = e(T_{B2}^*, T_{B2}^*)^{a_1 a_2} = e(cP, cP)^{a_1 a_2}$$
$$K_7 = e(T_{B1}^*, T_{B1}^*)^{a_2 a_2}$$
$$K_8 = e(T_{B1}^*, T_{B2}^*)^{a_2 a_2}$$
$$K_9 = e(T_{B2}^*, T_{B2}^*)^{a_2 a_2} = e(cP, cP)^{a_2 a_2}$$

Finally, the adversary E can use the random number c to compute three session keys

$$K_3 = e(T_{A1}, T_{A1})^{cc} = e(P,P)^{a_1 a_1 cc} = e(cP, cP)^{a_1 a_1}$$
$$K_6 = e(T_{A1}, T_{A2})^{cc} = e(P,P)^{a_1 a_2 cc} = e(cP, cP)^{a_1 a_2}$$
$$K_9 = e(T_{A2}, T_{A2})^{cc} = e(P,P)^{a_2 a_2 cc} = e(cP, cP)^{a_2 a_2}$$

Clearly, the adversary E has mounted the basic impersonation attack successfully and obtained three session keys. So the FAAJ protocol is not secure against the basic impersonation attack.

3.2 Attack 2

In this subsection, we present the second attack against the FAAJ protocol. Our attack is the combination of key compromise impersonation attack and parallel session attack.

We assume the malicious party C has obtained party A's private key. Then the malicious party C can mount the combination attack as follows:

Step 1: Party A randomly chooses $a_1, a_2 \in Z_q^*$, and computes $T_{A1} = a_1 P, T_{A2} = a_2 P$. If $k_{A1} \cdot k_{A2} \neq 0$, then A computes

$$S_A = (k_{A1} \cdot k_{A2}) x_A - (a_1 k_{A1} + a_2 k_{A2}) \bmod q .$$

Otherwise, A chooses a new random number and start the session again. Finally, A sends $\{T_{A1}, T_{A2}, S_A, Cert(Y_A)\}$ to party B .

Step 1*: Party C first intercepts $\{T_{A1}, T_{A2}, S_A, Cert(Y_A)\}$. Then C sets $T_{C1} = T_{A1}, T_{C2} = T_{A2}$, and computes

$$S_C = (k_{A1} \cdot k_{A2}) x_C + (S_A - k_{A1} \cdot k_{A2} \cdot x_A) \bmod q .$$

Finally, C sends $\{T_{C1}, T_{C2}, S_C, Cert(Y_C)\}$ to party B .

Step 2*: Upon receiving $\{T_{C1}, T_{C2}, S_C, Cert(Y_C)\}$, B verifies $Cert(Y_C)$. If it fails, B terminates the session. Otherwise, B checks the following equation

$$(k_{C1} \cdot k_{C2}) Y_C = S_C P + (k_{C1} T_{C1} + k_{C2} T_{C2}) ,$$

where $k_{C1} = k_{A1}, k_{C2} = k_{A2}$ due to $T_{C1} = T_{A1}, T_{C2} = T_{A2}$.

Since

$$\begin{aligned}
&S_C P + (k_{C1} T_{C1} + k_{C2} T_{C2}) \\
&= S_C P + (k_{A1} T_{A1} + k_{A2} T_{A2}) \\
&= [(k_{A1} \cdot k_{A2}) x_C + (S_A - k_{A1} \cdot k_{A2} \cdot x_A)] P + (k_{A1} T_{A1} + k_{A2} T_{A2}) \\
&= (k_{A1} \cdot k_{A2}) x_C P + (S_A - k_{A1} \cdot k_{A2} \cdot x_A) P + (k_{A1} T_{A1} + k_{A2} T_{A2}) \\
&= (k_{A1} \cdot k_{A2}) Y_C + S_A P - k_{A1} \cdot k_{A2} \cdot x_A P + (k_{A1} T_{A1} + k_{A2} T_{A2}) \\
&= (k_{A1} \cdot k_{A2}) Y_C + [(k_{A1} \cdot k_{A2}) x_A - (a_1 k_{A1} + a_2 k_{A2})] P - \\
&\qquad k_{A1} \cdot k_{A2} \cdot x_A P + (k_{A1} T_{A1} + k_{A2} T_{A2}) \\
&= (k_{A1} \cdot k_{A2}) Y_C + (k_{A1} \cdot k_{A2}) x_A P - (a_1 k_{A1} + a_2 k_{A2}) P \\
&\qquad + k_{A1} \cdot k_{A2} \cdot x_A P + (k_{A1} T_{A1} + k_{A2} T_{A2}) \\
&= (k_{A1} \cdot k_{A2}) Y_C - (k_{A1} T_{A1} + k_{A2} T_{A2}) + (k_{A1} T_{A1} + k_{A2} T_{A2}) \\
&= (k_{A1} \cdot k_{A2}) Y_C ,
\end{aligned}$$

the verification holds.

Then B will randomly choose $b_1, b_2 \in Z_q^*$ and compute $T_{B1} = b_1 P,\ T_{B2} = b_2 P$. If $k_{B1} \cdot k_{B2} \neq 0$, then B computes

$$S_B = (k_{B1} \cdot k_{B2}) x_B - (b_1 k_{B1} + b_2 k_{B2}) \bmod q .$$

Otherwise, B chooses a new random number and computes T_{B1}, T_{B2} again. Finally, B sends $\{T_{B1}, T_{B2}, S_B, Cert(Y_B)\}$ to party C and generates the session keys as follows:

$$\begin{aligned}
K_1 &= e(T_{C1}, T_{C1})^{b_1 b_1} = e(P,P)^{a_1 a_1 b_1 b_1} \\
K_2 &= e(T_{C1}, T_{C1})^{b_1 b_2} = e(P,P)^{a_1 a_1 b_1 b_2} \\
K_3 &= e(T_{C1}, T_{C1})^{b_2 b_2} = e(P,P)^{a_1 a_1 b_2 b_2} \\
K_4 &= e(T_{C1}, T_{C2})^{b_1 b_1} = e(P,P)^{a_1 a_2 b_1 b_1} \\
K_5 &= e(T_{C1}, T_{C2})^{b_1 b_2} = e(P,P)^{a_1 a_2 b_1 b_2} \\
K_6 &= e(T_{C1}, T_{C2})^{b_2 b_2} = e(P,P)^{a_1 a_2 b_2 b_2} \\
K_7 &= e(T_{C2}, T_{C2})^{b_1 b_1} = e(P,P)^{a_2 a_2 b_1 b_1} \\
K_8 &= e(T_{C2}, T_{C2})^{b_1 b_2} = e(P,P)^{a_2 a_2 b_1 b_2} \\
K_9 &= e(T_{C2}, T_{C2})^{b_2 b_2} = e(P,P)^{a_2 a_2 b_2 b_2}
\end{aligned}$$

Step 3*: Upon receiving $\{T_{B1}, T_{B2}, S_B, Cert(Y_B)\}$, party C impersonates party B and sends $\{T_{B1}, T_{B2}, S_B, Cert(Y_B)\}$ to party A .

Step 2: Upon receiving $\{T_{B1}, T_{B2}, S_B, Cert(Y_B)\}$, A verifies $Cert(Y_B)$. If it fails, A terminates the session. Otherwise, A checks the following equation

$$(k_{B1} \cdot k_{B2}) Y_B = S_B P + (k_{B1} T_{B1} + k_{B2} T_{B2}) .$$

Clearly, the verification will hold. So A generates the session keys as follows:

$$\begin{aligned}
K_1 &= e(T_{B1}, T_{B1})^{a_1 a_1} = e(P,P)^{a_1 a_1 b_1 b_1} \\
K_2 &= e(T_{B1}, T_{B2})^{a_1 a_1} = e(P,P)^{a_1 a_1 b_1 b_2} \\
K_3 &= e(T_{B2}, T_{B2})^{a_1 a_1} = e(P,P)^{a_1 a_1 b_2 b_2} \\
K_4 &= e(T_{B1}, T_{B1})^{a_1 a_2} = e(P,P)^{a_1 a_2 b_1 b_1} \\
K_5 &= e(T_{B1}, T_{B2})^{a_1 a_2} = e(P,P)^{a_1 a_2 b_1 b_2} \\
K_6 &= e(T_{B2}, T_{B2})^{a_1 a_2} = e(P,P)^{a_1 a_2 b_2 b_2} \\
K_7 &= e(T_{B1}, T_{B1})^{a_2 a_2} = e(P,P)^{a_2 a_2 b_1 b_1} \\
K_8 &= e(T_{B1}, T_{B2})^{a_2 a_2} = e(P,P)^{a_2 a_2 b_1 b_2} \\
K_9 &= e(T_{B2}, T_{B2})^{a_2 a_2} = e(P,P)^{a_2 a_2 b_2 b_2}
\end{aligned}$$

Now, the malicious party C has used party A's private key to make two different sessions generate the same session keys. Party A believes that he has completed a session and shared nine session keys with party B , who is not involved the session at all. At the same time, party B believes that he has completed a session and shared nine session keys with party C . In this attack, party C cannot compute the session keys. But party C can reveal the session keys shared between party B and himself to cheat party A . This will be a serious problem in practices.

4 Conclusions

In this letter, we have pointed out that Farash et al.'s protocol is insecure against the basic impersonation attack. We also show that Farash et al.'s protocol cannot resist the combination of key compromise impersonation attack and parallel session attack. To avoid these security flaws, it must be carefully design Farash et al.'s protocol again.

Acknowledgments

This study was supported by the Science Foundation of Luoyang University of Foreign Languages (2011XYZ004). The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] Q. Cheng and C. Ma, "Analysis and improvement of an authenticated multiple key exchange protocol," *Computers and Electrical Engineering*, vol. 37, no. 2, pp.187-190, 2011.

[2] M. Dehkordi and R. Alimoradi, "Identity-based multiple key agreement scheme," *KSII Transactions on Internet and Information Systems*, vol. 5, no. 12, pp. 2392-2402, 2011.

[3] M. Farash, M. Attari, R. Atani, and M. Jami, "A new efficient authenticated multiple-key exchange protocol from bilinear pairings," *Computers and Electrical Engineering*, vol. 39, no. 2, pp. 530-541, 2013.

[4] L. Harn and H. Lin, "An authenticated key agreement protocol without using one-way functions," in P*roceedings of 8th National conference on Information* Security, pp. 155-160, 1998.

[5] N. Lee, C. Wu, and C. Wang, "Authenticated multiple key exchange protocols based on elliptic curves and bilinear pairings," *Computers and Electrical Engineering*, vol. 34, no. 1, pp. 12-20, 2008.

[6] K. Shim, "Unknown key-share attack on authenticated multiple-key agreement protocol," *Electronics Letters*, vol. 39, no. 1, pp. 38-39, 2003.

[7] Z. Tan, "Efficient identity-based authenticated multiple key exchange protocol," *Computers and Electrical Engineering*, vol. 37, no. 2, pp. 191-198, 2011.

[8] Z. Tan, "Identity-based authenticated multiple key agreement protocol with PKG forward security," *KSII Transactions on Internet and Information Systems*, vol. 6, no. 8, pp. 1982-1997, 2012.

[9] D. Vo, H. Lee, C. Yeun, and K. Kim, "Enhancements of authenticated multiple key exchange protocol based on bilinear pairings," *Computers and Electrical Engineering*, vol. 36, no. 1, pp. 155-159, 2010.

[10] T. Wu, W. He, and C. Hsu, "Security of authenticated multiple-key agreement protocols," *Electronic Letters*, vol. 35, no. 5, pp. 391-392, 1999.

[11] S. Yen and M. Joye, "Improved authenticated multiple-key agreement protocol," *Electronics Letters*, vol. 34, no. 18, pp. 1738-1739, 1998.

Qingfeng Cheng received his B.A. degree in 2000 and M.S. degree in 2004 from National University of Defense Technology, and Ph.D. degree in 2011 from Information Engineering University. He is now an Associate Professor with the Department of Language Engineering, Luoyang University of Foreign Languages. His research interests include cryptography and information security.