Security flaws in a biometrics-based multi-server authentication with key agreement scheme

Debiao He*

School of Mathematics and Statistics, Wuhan University, Wuhan, People's Republic of China

Email: hedebiao@whu.edu.cn

Abstract: Recently, Yoon et al. proposed an efficient biometrics-based multi-server authentication with key agreement scheme for smart cards on elliptic curve cryptosystem (ECC) for multi-server communication environments [E.-J. Yoon, K.-Y. Yoo(2011) Robust biometrics-based multi-server authentication with key agreement scheme for smart cards on elliptic curve cryptosystem, Journal of Supercomputing, DOI: 10.1007/s11227-010-0512-1]. They claimed their scheme could withstand various attacks. In the letter, we will show Yoon et al.'s scheme is vulnerable to the privileged insider attack, the masquerade attack and the smart cart lost attack.

***Key words**: Authentication; Key agreement; Masquerade attack; Privileged insider attack; Elliptic curve cryptosystem; Smart card*

1. Introduction

Following the advances in network technologies and the widespread distribution of remote system backup, lots of multi-server based applications have been deployed to make legitimate user access network service (or resource) more conveniently and efficiently. Primarily via the Internet, facilities and computers are linked together and the resource can be easily shared and exploited. As a result, an adequate remote user verification procedure must be adopted to ensure legal resource access and secure data exchange. As a password based user authentication scheme provides an efficient and accurate way to identify valid remote user and at the same time preserves the secrecy of communication, various authentication mechanisms for single-server environment have been investigated in recent years. However, these single-server authentication schemes suffer a significant shortcoming. If a remote user wishes to use numerous network services, they must register their identity and password at these servers. It is extremely tedious for users to register numerous servers.

Recently. Yoon et al.[1] also proposed a new efficient and secure biometrics-based multi-server authentication with key agreement scheme for smart cards on elliptic curve cryptosystem (ECC) without a verification table to minimize the complexity of hash operation among all users and fit multi-server communication environments. They claimed that their scheme is secure against various attacks. However, in this letter, we show that Yoon et al.'s scheme can't resist the impersonation attack, the insider's attack and the smart card lost attack. We also proposed countermeasures to withstand the attacks.

The rest of the letter is organized as follows. Section 2 gives the review of the Yoon et al.'s scheme. Section 3 discusses the cryptanalysis of Yoon et al.'s scheme. In Section 4, two efficient countermeasures are proposed. Finally, we conclude the paper in Section 5.

2. Review of Yoon et al.'s scheme

Before the review of Yoon et al.'s scheme [1], we first introduce some notations as follows, which are common with that used in [1].

- U_i, S_j : the i th user and j th server, respectively;
- RC : the registration center;
- ID_i, PW_i, B_i : U_i 's identity, password and biometric template, respectively;
- SID_j : S_j 's identity;
- x : U_i 's secret key maintained by the registration center;
- y : S_j 's secret key maintained by the registration center;
- n, p : large prime number;
- F_p : finite prime field;
- E : non-super singular Elliptic curve over a finite field F_p, where E : $y^2 = x^3 + ax + b \bmod p$ with $a, b \in F_p$ satisfying $4a^3 + 27b \neq 0 \bmod p$;
- G : additive group of points on E over a finite field F_p, where $G = \{(x, y) \mid x, y \in F_p, y^2 = x^3 + ax + b \bmod p\} \cup \{O\}$ and the order of G is n;
- P : generating element (point) of G;

- α, β : session-independent random integer numbers chosen by U_i and S_j, respectively;
- SK : shared fresh session key computed by U_i and S_j;
- $d(\cdot)$: symmetric parametric function;
- τ : predetermined threshold for biometric verification;
- $h(\cdot)$: secure one-way hash function;
- $\oplus$:bit-wise exclusive-or(XOR) operation;
- $||$: concatenation operation;
- D : a uniformly distributed dictionary of size $|D|$;

The proposed scheme is composed of four phases, which are the server registration phase, the user registration phase, the authenticated key agreement phase, and the password and biometrics update phase. The detail is described as follows.

2.1. Server registration phase

When a server S_j wants to register and become a new legal server, the following steps will be executed.

$SR.1$. S_j freely chooses his identity SID_j and submits it to RC via secure channel.

$SR.2$. Upon receiving SID_j, RC computes $R_j = h(SID_j || y)$, where y is a S_j's secret key maintained by RC, and sends it to S_j via secure channel.

$SR.3$. Upon receiving R_j, S_j stores it secretly and finishes the registration.

2.2. User registration phase

When a user U_i wants to register and become a new legal user, the following steps are performed during the user registration phase.

$UR.1$. U_i freely chooses his identity ID_i, password PW_i, and also imprints his personal biometric impression B_i at the sensor. U_i then interactively submits $\{ID_i, B_i, h(PW_i || B_i)\}$ to RC via secure channel.

$UR.2$ RC computes $R_i = h(ID_i || x)$ and $Z_i = R_i \oplus h(PW_i || B_i)$, where x is a U_i's secret key maintained by RC. Then, RC writes the secure information

$\{Z_i, B_i, h(\cdot), d(\cdot), \tau\}$ to the memory of U_i's smart card and issues it to U_i through a secure channel, where $d(\cdot)$ is a symmetric parametric function and τ is a predetermined threshold [2] for biometric verification.

2.3. Authenticated key agreement phase

When U_i wants login in S_j, the following steps are performed during the authenticated key agreement phase.

$A.1$. U_i inserts his smart card into a card reader, opens the login application software, and imprints biometric B_i^* at the sensor. Then a biometric verification process of U_i's smart card compares the imprinted B_i^* with the stored B_i. If $d(B_i, B_i^*) \geq \tau$, U_i's smart card rejects the request. Otherwise, U_i enters his password PW_i and his identity ID_i, and then the smart card generates a random integer number $\alpha \in [1, n-1]$, computes $R_i = Z_i \oplus h(PW_i \| B_i)$, $X = \alpha P$ and $C_1 = h(R_i \| X)$. Then U_i sends $M_1 = \{ID_i, X, C_1\}$ to S_j.

$A.2$. Upon receiving the message M_1, S_j generates a random integer number $\beta \in [1, n-1]$ and computes $Y = \beta P$, $C_2 = h(R_j \| Y)$ and sends the message $M_2 = \{ID_i, X, C_1, SID_j, Y, C_2\}$ to RC.

$A.3$. Upon receiving the message M_2, RC computes $C_1' = h(h(ID_i \| x) \| X)$ and $C_2' = h(h(SID_j \| y) \| Y)$ and checks whether C_1 and C_2 equal C_1' and C_2' respectively. If not, RC stops the session. Otherwise, RC computes $V = h(h(SID_j \| y) \| Y \| X)$, $W = h(h(ID_i \| x) \| SID_j \| X \| Y)$, $C_3 = V \oplus W$, $C_4 = h(V \| W)$. At last, RC sends the message $M_3 = \{C_3, C_4\}$ to S_j.

$A.4$. Upon receiving the message M_3, S_j computes $V' = h(R_j \| Y \| X)$, $W' = V' \oplus C_3$ and $C_4' = h(V' \| W')$. Then S_j checks whether C_4' and C_4 are equal. If not, S_j stops the session. Otherwise, S_j computes the session key $SK_j = \beta X = \alpha\beta P$ and $C_5 = h(ID_i \| SID_j \| W \| SK_j)$. Finally, S_j sends $M_4 = \{Y, C_5\}$ to U_i.

$A.5$. Upon receiving the message M_4 , U_i computes $W'' = h(R_i \| SID_j \| X \| Y)$, $SK_i = \alpha Y = \alpha\beta P$ and $C_5' = h(ID_i \| SID_i \| W'' \| SK_i)$. Then U_i checks whether C_5' and C_5 are equal. If not, U_i stops the session. Otherwise, U_i computes $C_6 = h(W'' \| SK_i \| Y)$ and sends $M_5 = \{C_6\}$ to S_j .

$A.6$. Upon receiving the message M_5, S_j computes $C_6' = h(W' \| SK_j \| Y)$ and checks whether C_6' and C_6 are equal. If they are equal, S_j confirms the legality of U_i. Otherwise, S_j stops the session.

2.4. Password and biometrics update phase

In this phase, the user U_i can freely and securely change the old password PW_i to a new password PW_i' and the old biometrics B_i to a new biometrics B_i', respectively, without helping of the registration center RC. The biometrics update requires because the biometrics has the problem of the aged deterioration.

$P.1$. U_i inserts his smart card into a card reader, opens the login application software, and imprints biometric B_i' at the sensor.

$P.2$. Then a biometric verification process of U_i's smart card compares the imprinted B_i' with the stored B_i. If $d(B_i, B_i') \geq \tau$, U_i's smart card rejects the request. Otherwise, U_i's smart card shows a password input request message to the user U_i.

$P.3$. U_i enters his old password PW_i and a new password PW_i'.

$P.4$. U_i 's smart card computes $Z_i' = Z_i \oplus h(PW_i \| B_i) \oplus h(PW_i' \| B_i')$ and replace Z_i with Z_i'.

3. Weakness in Yoon et al.'s scheme

3.1. Privileged Insider Attack

In a real environment, it is a common practice that many users use same passwords to access different applications or servers for their convenience of remembering long passwords and ease-of-use whenever required. However, if the system manager or a privileged-insider of the RC knows the passwords of the user U_i, he may try to impersonate U_i by accessing other servers where U_i could be a registered user. In the user registration phase of Yoon et al.'s scheme, U_i sends his identity ID_i, biometric impression B_i and $h(PW_i \| B_i)$ to RC. Although, the password PW_i is not directly transmitted to the system, the privileged-insider of the RC could get the password through the off-line password guessing attack. The detail of the off-line password guessing attack is described as follows.

1). The privileged-insider A chooses guess a password PW_i' from D.

2). A computes $h_1 = h(PW_i' \| B_i)$.

3). A verifies whether $h(PW_i \| B_i)$ and h_1 are equal. If $h(PW_i \| B_i)$ and h_1 are equal, the adversary gets the correct password. Otherwise, A repeats Step 1, Step 2 and Step 3 in the second phase until finding the correct password.

From the above description, we know the adversary can get the password. Therefore, Yoon et al.'s scheme is vulnerable to the privileged insider attack.

3.2. Masquerade attack

We assume that an attacker A has total control over the communication channel among the user U_i, the remote server S_j and the registration center RC, which means that he can insert, delete, or alter any messages in the channel. We shall prove that Yoon et al.'s scheme cannot withstand the masquerade attack, if A is a legal user of the system. The adversary A can masquerade as any legal user U_i to login the remote server S_j without knowing the password PW_i at anytime. He can forge a login message that can pass S_j's authentication. A more detailed description of the attack is as follows.

1) A generates a random integer number $\alpha \in [1, n-1]$, computes $R_A = Z_A \oplus h(PW_A \| B_A)$, $X = \alpha P$ and $C_1 = h(R_A \| X)$. Then A sends $M_1 = \{ID_i, X, C_1\}$ to S_j.

2) Upon receiving the message $M_1 = \{ID_i, X, C_1\}$, S_j generates a random integer number $\beta \in [1, n-1]$ and computes $Y = \beta P$, $C_2 = h(R_j \| Y)$ and sends the message $M_2 = \{ID_i, X, C_1, SID_j, Y, C_2\}$ to RC.

3) A intercepts the message M_2 and sends the messages $M_2' = \{ID_A, X, C_1, SID_j, Y, C_2\}$

4) Upon receiving the message M_2', RC computes $C_1' = h(h(ID_A \| x) \| X)$ and $C_2' = h(h(SID_j \| y) \| Y)$ and checks whether C_1 and C_2 equal C_1' and C_2' respectively. If not, RC stops the session. Otherwise, RC computes $V = h(h(SID_j \| y) \| Y \| X)$, $W = h(h(ID_A \| x) \| SID_j \| X \| Y)$, $C_3 = V \oplus W$, $C_4 = h(V \| W)$. At last, RC sends the message $M_3 = \{C_3, C_4\}$ to S_j.

5) Upon receiving the message M_3, S_j computes $V' = h(R_j \| Y \| X)$, $W' = V' \oplus C_3$ and $C_4' = h(V' \| W')$. Then S_j checks whether C_4' and C_4 are equal. It is easy to say C_4' and C_4 are equal. Then S_j computes the session key $SK_j = \beta X = \alpha\beta P$ and $C_5 = h(ID_i \| SID_j \| W \| SK_j)$. Finally, S_j sends $M_4 = \{Y, C_5\}$ to A.

6) Upon receiving the message M_4, A computes $W'' = h(R_A \| SID_j \| X \| Y)$, $SK_i = \alpha Y = \alpha\beta P$ and $C_5' = h(ID_i \| SID_j \| W'' \| SK_i)$. A computes $C_6 = h(W'' \| SK_i \| Y)$ and sends $M_5 = \{C_6\}$ to S_j.

7) Upon receiving the message M_5, S_j computes $C_6' = h(W' \| SK_j \| Y)$ and checks whether C_6' and C_6 are equal. It is easy to say C_6' and C_6 are equal. Then the adversary A impersonates the legal user U_i successfully.

4.3. Stolen smart card attack

Yoon et al. claimed that their scheme could resist stolen smart card attack. However, in this section, we will show their scheme is still vulnerable to the stolen smart card attack.

To evaluate the security of smart card based user authentication, many researchers assume that the capabilities that an adversary A may have as follows:

(1) The adversary has total control over the communication channel between the users and the server in the login and authentication phases. That is, A may intercept, insert, delete, or modify any message in the channel.

(2) A may (i) either steal a user's smart card and then extract the information from it, (ii) or obtain a user's password, (iii) but not both (i) and (ii).

Kocher et al. [13] and Messerges et al. [14] have pointed out that all existent smart cards are vulnerable in that the confidential information stored in the device could be extracted by physically monitoring its power consumption; once a card is lost, all secrets in it may be revealed. It is trivial to see that if a user's smart card and his password are both stolen, there is no means to prevent the adversary from masquerading as the user. In this paper, we are especially interested in the security of password authentication schemes in the case that the smart card is stolen but the user password of the device owner is unknown to the adversary.

We assume that the user U_i's smart card is lost and the adversary A get it. Then A could read all the sensitive information $\{Z_i, B_i, h(\cdot), d(\cdot), \tau\}$ from the smart card by executing side channel attack[23, 24], where $Z_i = R_i \oplus h(PW_i \| B_i)$ and $R_i = h(ID_i \| x)$. We also assume that A knows the identity ID_i of the U_i. Then obtains the message $M_1 = \{ID_i, X, C_1\}$ generated in some previous session according to ID_i, where $X = \alpha P$, $C_1 = h(R_i \| X)$ and $R_i = Z_i \oplus h(PW_i \| B_i)$. A could carry out the off-line password guessing attack as follows.

1) A guesses a password PW_i^* from D;

2) A computes $R_i^* = Z_i \oplus h(PW_i^* \| B_i)$, $C_1^* = h(R_i^* \| X)$ and checks whether C_1^* and C_1 are equal. If they are equal, A finds the correct password. Otherwise, A repeats 1) and 2) until finding the correct password.

Our attack is feasible because both password and identity are human-memorable short strings but not high-entropy keys. In other words, they are chosen from the two corresponding dictionaries of small size. In addition, the attacker can probably deduce the user’s identity when she gets the smart card. In that case, our attack can be done much more efficiently since she only needs to guess the password. This assumption is reasonable because the user often chooses his name as his identity or write his identity on the card; and moreover the input identity is usually displayed in plain on the screen and thus can be possibly seen when the attacker steals the card. After all, the attack can know more or less about the personal information of the card holder when she steals the card.

After she has obtained the correct password PW_i and identity ID_i, she also knows the secret value of R_i by computing $R_i = Z_i \oplus h(PW_i \| B_i)$. As a result, the attacker can impersonate U_i to login successfully. In a word, the adversary will be able to break the scheme completely if the smart-cart is compromised and the secrets stored in it are revealed.

5. Conclusion

In this paper, we first reviewed Yoon et al.'s scheme [11], and pointed out that their scheme can't resist the privileged insider attack, the masquerade attack and the smart card lost attack.

Reference

[1]. E.-J. Yoon, K.-Y. Yoo(2011) Robust biometrics-based multi-server authentication with key agreement scheme for smart cards on elliptic curve cryptosystem, Journal of Supercomputing, DOI: 10.1007/s11227-010-0512-1

[2]. Inuma M, Otsuka A, Imai H (2009) Theoretical framework for constructing matching algorithms in biometric authentication systems. In: Proc of ICB’09. Lecture notes in computer science, vol 5558. Springer, Berlin, pp 806–815