Cryptanalysis of C-3 PEKE Protocol

R.Padmavathy
National Institute of
Technology, Warangal,
Andhra Pradesh, India

ABSTRACT
The key exchange protocol using passwords achieved great attention due to its simplicity and efficiency. Recently, Chang proposed a practical three-party key exchange (C-3 PEKE) protocol with round efficiency. Later, Lee and Chang presented an off-line password guessing attack on C-3 PEKE protocol. In the present paper, an impersonation-of-the initiator attack and impersonation-of-the responder attack are demonstrated on C-3 PEKE protocol using the off-line password guessing attack proposed by Lee and Chang.

.**Keywords-** C-3 PEKE protocol, off-line password guessing attack, impersonation-of-the-initiator attack, impersonation-of-the-responder attack

1. INTRODUCTION
The key exchange protocol is one of the most elegant ways of establishing secure communication between pair of users by using a session key. The session key, which is exchanged between two users, assures the secure communication for later sessions. The first practical key exchange protocol is proposed by Diffie-Hellman [10]. Since the introduction of key ex-change protocol by Diffie-Hellman, various versions and improvements in key exchange protocol have been developed [1,4,6,7,12]. In the line of key exchange protocol development, password based key exchange mechanism achieved great attention due to its simplicity and wide range of applicability, as it requires the users to remember the easily remembrable password. Even though the protocol is simple and efficient, it should not be vulnerable to any type of off line, undetectable or detectable on line password guessing attacks, since the passwords are of low-entropy.

In general the password guessing attacks can be divided into three classes and they are listed below:

• **Detectable on-line password guessing attacks**: An attacker attempts to use a guessed password in an on-line transaction. He/She verifies the correctness of his/her guess using the response from server. A failed guess can be detected and logged by the server.

• **Undetectable on-line password guessing attacks**: Similar to Detectable on-line password guessing attack, an attacker tries to verify a password guess in an on-line transaction. However, a failed guess cannot be detected and logged by server, as server is not able to distinguish an honest request from a malicious one.

• **Off-line password guessing attacks**: An attacker guesses a password and verifies his/her guess off-line. No participation of server is required, so the server does not notice the attack.

Since the first proposal of Bellovin and Merrit (PAKE) [5], many efficient key exchange protocols based on password have been developed. Recently these two Party key exchange protocols are extended to three party [2, 3, 7, 11, 12, 13, 14, 15, 16, 17, 18], in which, the two parties initially communicates the passwords with the trusted server securely. Later the server authenticates the clients when they want to agree upon a session key. The 3-party protocol is introduced by Steiner et al [19]. Subsequently Ding and Hoster presented on line and offline guessing attacks on Stener's protocol [9].

Recently, Chang proposed a practical three-party key exchange (C-3 PEKE) protocol with round efficiency [8]. Later, Lee and Chang presented an Off-line password guessing attack on C-3 PEKE protocol [13]. In the present paper, an impersonation -of-the initiator attack and impersonation-of-the responder attack are demonstrated on C-3 PEKE protocol using the off-line password guessing attack proposed by Lee and Chang.

The rest of the paper is organized as follows: Section2 briefly describes C-3 PEKE (Chang's three-party key exchange protocol) and off-line password guessing attack on C-3 PEKE protocol proposed by Lee and Chang. Section 3 demonstrates impersonation-of-initiator attack and impersonation-of-the responder attack on C-3 PEKE protocol and the concluding remarks are made in section 4.

2. REVIEW OF C-3 PEKE PROTOCOL

This section presents C-3 PEKE (Chang's three-party key exchange protocol).

Notations

A,B: Communication parties

S: the trusted Server

$ID_A/ID_B/ID_S$: the identities of A/B/S

P_A/P_B: the password securely shared by A/B with S

$E_{3p}( )$: a symmetric encryption scheme with a password p

P: a large prime

g: an element of order q with modulus p

G: a finite cyclic group generated by g in Z_P

R_A/R_B: the random exponents chosen by A/B

R_{S1}/R_{S2}: two random exponents chosen by S

N_A, N_B: $N_A=g^{RS1}$modp, $N_B=g^{RS2}$modp

N_{S1}, N_{S2}: $N_{S1}=g^{RS1}$modp, $N_{S2}=g^{RS2}$modp

$f_K(.)$: a pseudo-random function (PRF) indexed by K

K_{AS}/K_{BS}: a one-time key shared key by A/B and S

K_{AB}: a session key shared by A and B

Step1: A selects a random number R_A and computes $N_A = g^{R_A} \bmod p$ and the sends ID_A, ID_B, N_A to *S* as request.

Step2: After receiving A's request, S chooses two random numbers R_{S_1}, R_{S_2} and computes $N_{S_1} = g^{RS_1} \bmod p$, $N_{S_2} = g^{RS_2} \bmod p$ then determines $K_{AS} = N_A^{RS_1} \bmod p = g^{RS_1 R_A} \bmod p$. Then, *S* sends $\left(ID_A, N_A, E_{3P_A}\left(N_{S_1}\right), E_{3P_B}\left(N_{S_2}\right), f_{K_{AS}}\left(ID_A, ID_B, N_A\right)\right)$ to *B*

Step3: Upon receiving *S*'s message, *B* first decrypts $E_{3P_B}\left(N_{S_2}\right)$ using password p_B, to get N_{S_2}. Then *B* computes $K_{BS} = N_{S_2}^{R_B} \bmod p = g^{RS_2 R_B} \bmod p$ and $K_{AB} = N_A^{R_B} \bmod p = g^{R_A R_B} \bmod p$. Next, *B* sends $\left(ID_B, N_B, E_{3P_A}\left(N_{S_1}\right), f_{K_{AS}}\left(ID_A, ID_B, N_A\right), f_{K_{AB}}\left(ID_A, ID_B, N_A\right), f_{K_{BS}}\left(ID_A, ID_B, N_{S_2}\right)\right)$ to *A*.

Step4: Upon receiving the message, *A* first decrypts $E_{3P_A}\left(N_{S_1}\right)$using P_A to get N_{S_1}. Then *A* computes $K_{AS} = N_{S_1}^{R_A} \bmod p = g^{RS_1 R_A} \bmod p$ and $K_{AB} = N_B^{R_A} \bmod p = g^{R_A R_B} \bmod p$. Firstly, *A* uses K_{AS} to compute $f_{K_{AS}}\left(ID_A, ID_B, N_A\right)$ and verifies if the computation result is equal to the received one. If it is correct, *A* believes that he/she is communicating with a legitimate *S*; otherwise, *A* regards *S* illegal and terminates the protocol. Secondly, *A* uses K_{AB} to compute $f_{K_{AB}}\left(ID_A, ID_B, N_A\right)$ and verifies if the computation result is equal to the received one. If it is correct, *A* believes that he/she is communicating with a legitimate *B*; otherwise, *A* regards *B* illegal and terminates the protocol. After authenticating *S* and *B*, *A* sends $(ID_A, ID_B, N_B, f_{K_{AS}}\left(ID_A, ID_B, N_{S_1}\right), f_{K_{BS}}\left(ID_A, ID_B, N_{S_2}\right), f_{K_{AB}}\left(ID_A, ID_B, N_B\right))$ to *S*.

Step 5: Upon receiving the message, *S* computes $K_{BS} = N_B^{RS_2} \bmod p = g^{RS_2 R_B} \bmod p$. Firstly, *S* uses K_{AS} to compute $f_{K_{AS}}\left(ID_A, ID_B, N_{S_1}\right)$ and verifies if the computation result is equal to the received one. If it is correct, *S* believes that he/she is communicating with a legitimate *A*; otherwise, S regards *A* illegal and terminates the protocol. Secondly, *S* uses K_{BS} to compute $f_{K_{BS}}\left(ID_A, ID_B, N_{S_2}\right)$ and verifies if the computation result is equal to the received one. If it is correct, *S* believes that he/she is communicating with a legitimate *B*; otherwise, *S* regards *B* illegal and terminates the protocol. After authenticating *A* and *B*, *S* sends $\left(ID_A, f_{K_{AB}}\left(ID_A, ID_B, N_B\right), f_{K_{BS}}\left(ID_A, ID_B, N_B\right)\right)$ to *B*. After receiving the message, *B* uses K_{BS} to compute $f_{K_{BS}}\left(ID_A, ID_B, N_B\right)$ and verifies if the computation result is equal to the received one. If it is correct, *B* believes that he/she is communicating with a legitimate *S*; otherwise, *B* regards *S* illegal and terminates the protocol. Next, *B* uses K_{AB} to compute $f_{K_{AB}}\left(ID_A, ID_B, N_B\right)$ and verifies if the computation result is equal to the received one. If it is correct, *B* believes that he/she is communicating with a legitimate *A*; otherwise, *B* regards *A* illegal and terminates the protocol.

Finally, *A* and *B* can share the session key K_{AB} to encrypt and decrypt their communicated messages. Fig 1 illustrates the C-3 PEKE protocol

3. OFF-LINE PASSWORD GUESSING ATTACK ON C-3 PEKE PROTOCOL

An attacker C can intercept transmitted messages from public channel and then break password by playing off-line guessing attacks. *C* can guess a password *P*' until the guessing *P*' is equal to the correct password *P*. Otherwise, *C* repeatedly guesses a new *P*' off-line. Suppose that *C* tends to get *A*'s password P_A, then the procedure followed is:

Step1: *C* wiretaps that *A* and *B* communicate with *S*. *C* can intercept ID_A and ID_B.

Step2: *C* forges *A* communicate with *S*. He/She chooses a new random number R'_A and computes $N'_A = g^{R'_A} \bmod p$. Then *C* forges *A* to send $\left(ID_A, ID_B, N'_A\right)$ to *S*.

Step3: After receiving C'_S request, *S* computes $K_{AS} = N'^{RS_1}_A \bmod p = g^{RS_1R'_A} \bmod p$. Then, *S* sends $ID_A, N'_A, E_{3P_A}(N_{S_1}), E_{3P_B}(N_{S_2}), f_{K_{AS}}(ID_A, ID_B, N'_A)$ to *B*. Since *C* wiretaps their communications, he/she can intercept $E_{3P_A}(N_{S_1})$ and $f_{K_{AS}}(ID_A, ID_B, N'_A)$.

Step4: Once C intercepts $E_{3P_A}(N_{S_1})$ and $f_{K_{AS}}(ID_A, ID_B, N'_A)$, he/she can play off-line guessing attacks.

Now *C* guesses a password P'. He/She first decrypts $E_{3P_A}(N_{S_1})$ using P'. If $P' = P_A$, he/she can get N_{S_1}. Then *C* can compute $K_{AS} = N_{S_1}^{R'_A} \bmod p = g^{RS_1R_A} \bmod p$. Next, *C* computes $f_{K_{AS}}(ID_A, ID_B, N'_A)$ and verifies if the computation result is equal to the intercepted one. If it is correct, *C* believes that he/she had guessed a correct password P'_A; otherwise, *C* repeatedly guesses a new P' off-line till *C* can guess a correct password P_A. In the same way, *C* can get *B*'s password P_B.

4. IMPERSONATION- OF- THE-INITIATOR ATTACK

This section presents an impersonation-of-initiator attack on Chang's three-party key exchange protocol C-3PEKE. Fig 2 illustrates the impersonation-of-initiator attack.

Step1: C selects a random number R_C and computes $N_C = g^{R_C} \bmod p$ and sends (ID_A, ID_B, N_C) to *S* (as *A*) as request.

Step2: After receiving *A*'s request, *S* chooses two random numbers R_{S_1}, R_{S_2} and computes $N_{S_1} = g^{RS_1} \bmod p$, $N_{S_2} = g^{RS_2} \bmod p$ then determines $K_{CS} = N_C^{RS_1} \bmod p = g^{RS_1R_C} \bmod p$. Then, *S* sends $(ID_A, N_C, E_{3P_A}(N_{S_1}), E_{3P_B}(N_{S_2}), f_{K_{CS}}(ID_A.ID_B, N_C))$ to B.

Step3: *B* selects a random number R_B and computes $N_B = g^{R_B} \bmod p$. Upon receiving *S*'s message, *B* first decrypts $E_{3P_B}(N_{S_2})$ using password P_B, to get N_{S_2}. Then *B* computes $K_{BS} = N_{S_2}^{R_B} \bmod p = g^{RS_2R_B} \bmod p$ and $K_{CB} = N_C^{R_B} \bmod p = g^{R_CR_B} \bmod p$. Next, *B* sends $(ID_B, N_B, E_{3P_A}(N_{S_1}), f_{K_{CS}}(ID_A.ID_B, N_C), f_{K_{CB}}(ID_A.ID_B, N_C)$, $f_{K_{BS}}(ID_A.ID_B, N_{S_2}))$ to A.

Step4: *C* intercepts this message, upon receiving the message, *C* first decrypts $E_{3P_A}(N_{S_1})$ using P_A (P_A is already determined as shown in section 2 (*A*)) to get N_{S_1}. Then *C* computes $K_{CS} = N_{S_1}^{R_C} \bmod p = g^{RS_1R_C} \bmod p$ and $K_{CB} = N_B^{R_C} \bmod p = g^{R_BR_C} \bmod p$ Firstly, *A* uses K_{CS} to compute $f_{K_{CS}}(ID_A, ID_B, N_C)$ and verifies if the computed result is equal to the received one. If it is correct, *C* believes that he/she is communicating with a legitimate *S*; otherwise, *C* regards *S* illegal and terminates the protocol. Secondly, *C* uses K_{CB} to compute $f_{K_{CB}}(ID_A, ID_B, N_C)$ and verifies if the computation result is equal to the received one. If it is correct, *C* believes that he/she is communicating with a legitimate *B*; otherwise, *C* regards *B* illegal and terminates the protocol. After authenticating *S* and *B*, *C* sends $(ID_A, ID_B, N_B, f_{K_{CS}}(ID_A, ID_B, N_{S_1}), f_{K_{BS}}(ID_A, ID_B, N_{S_2}), f_{K_{CB}}(ID_A, ID_B, N_B))$ to *S*.

Step 5: Upon receiving the message, *S* computes $K_{BS} = N_B^{RS_2} \bmod p = g^{RS_2R_B} \bmod p$. Firstly, *S* uses K_{CS} to compute $f_{K_{CS}}(ID_A, ID_B, N_{S_1})$ and verifies if the computation result is equal to the received one. If it is correct, *S* believes that he/she is communicating with a legitimate A **(But *S* is actually communicating with *C*)**; otherwise, *S* regards *A* illegal and terminates the protocol. Secondly, *S* uses K_{BS} to compute $f_{K_{BS}}(ID_A, ID_B, N_{S_2})$ and verifies if the computation result is equal to the received one. If it is correct, *S* believes that he/she is communicating with a legitimate *B*; otherwise, *S* regards *B* illegal and terminates the protocol. After authenticating *A* and *B*, *S* sends $(ID_A, f_{K_{CB}}(ID_A, ID_B, N_B), f_{K_{BS}}(ID_A, ID_B, N_B))$ to *B*. After receiving the message, *B* uses K_{BS} to compute $f_{K_{BS}}(ID_A, ID_B, N_B)$ and verifies if the computation result is equal to the received one. If it is correct, *B* believes that he/she is communicating with a legitimate *S*; otherwise, *B* regards *S* illegal and terminates the protocol. Next, *B* uses K_{CB} to compute $f_{K_{CB}}(ID_A, ID_B, N_B)$ and verifies if the computation result is equal to the received one. If it is correct, *B* believes that he/she is communicating with a legitimate *A* (But *B* is communicating with *C*); otherwise, *B* regards *A* illegal and terminates the protocol.

Finally, *C* and *B* can share the session key K_{CB} to encrypt and decrypt their communicated messages. *B* is thinking that it is communicating with *A* but actually it is communicating with *C*.

5. IMPERSONATION- OF- THE RESPONDER ATTACK

This section presents an impersonation-of-the responder attack on C-3PEKE (Chang's three-party key exchange protocol). Fig 3 illustrates the above attack.

Step1: A selects a random number R_A and computes $N_A = g^{R_A} \bmod p$ and the sends (ID_A, ID_B, N_A) to *S* as request.

Step2: After receiving *A*'s request, *S* chooses two random numbers R_{S_1}, R_{S_2} and computes $N_{S_1} = g^{RS_1} \bmod p$, $N_{S_2} = g^{RS_2} \bmod p$ then determines $K_{AS} = N_A{}^{RS_1} \bmod p = g^{RS_1R_A} \bmod p$. Then, S sends $(ID_A, N_A, E_{3P_A}(N_{S_1}), E_{3P_B}(N_{S_2}), f_{K_{AS}}(ID_A.ID_B, N_A))$ to B.

Step3: *C* selects a random number R_C and computes $N_C = g^{R_C} \bmod p$. Upon receiving *S*'s message, *C* first decrypts $E_{3P_B}(N_{S_2})$ using password P_B, (password of *B* is obtained by *C* as shown in section 2(*A*)) to get N_{S_2}. Then *C* computes $K_{CS} = N_{S_2}{}^{R_C} \bmod p = g^{RS_2R_C} \bmod p$ and $K_{AC} = N_A{}^{R_C} \bmod p = g^{R_AR_C} \bmod p$. Next, *C* sends $(ID_B, N_C, E_{3P_A}(N_{S_1}), f_{K_{AS}}(ID_A.ID_B, N_A), f_{K_{AC}}(ID_A.ID_B, N_A), f_{K_{CS}}(ID_A.ID_B, N_{S_2}))$ to A .

Step4: Upon receiving the message, *A* first decrypts $E_{3P_A}(N_{S_1})$ using P_A to get N_{S_1}. Then *A* computes $K_{AS} = N_{S_1}{}^{R_A} \bmod p = g^{RS_2R_A} \bmod p$ and $K_{AC} = N_C{}^{R_A} \bmod p = g^{R_AR_C} \bmod p$ Firstly, *A* uses K_{AS} to compute $f_{K_{AS}}(ID_A, ID_B, N_A)$ and verifies if the computation result is equal to the received one. If it is correct, *A* believes that he/she is communicating with a legitimate *S*; otherwise, *A* regards *S* illegal and terminates the protocol. Secondly, *A* uses K_{AC} to compute $f_{K_{AC}}(ID_A, ID_B, N_A)$ and verifies if the computation result is equal to the received one. If it is correct, *A* believes that he/she is communicating with a legitimate *B* (But actually *A* is communicating with *C*); otherwise, *A* regards *B* illegal and terminates the protocol. After authenticating *S* and *B*, *A* sends $(ID_A, ID_B, N_C, f_{K_{AS}}(ID_A, ID_B, N_{S_1}), f_{K_{CS}}(ID_A, ID_B, N_{S_2}), f_{K_{AC}}(ID_A, ID_B, N_C))$ to S.

Step 5: Upon receiving the message, *S* computes $K_{CS} = N_C{}^{RS_2} \bmod p = g^{RS_2R_C} \bmod p$. Firstly, *S* uses K_{AS} to compute $f_{K_{AS}}(ID_A, ID_B, N_{S_1})$ and verifies if the computation result is equal to the received one. If it is correct, S believes that he/she is communicating with a legitimate *A*; otherwise, *S* regards *A* illegal and terminates the protocol. Secondly, *S* uses K_{CS} to compute $f_{K_{CS}}(ID_A, ID_B, N_{S_2})$ and verifies if the computation result is equal to the received one. If it is correct, *S* believes that he/she is communicating with a legitimate *B* (But S is actually communicating with *C*); otherwise, *S* regards *B* illegal and terminates the protocol. After authenticating *A* and *B*, *S* sends $(ID_A, f_{K_{AB}}(ID_A, ID_B, N_B), f_{K_{BS}}(ID_A, ID_B, N_B))$ to *B*. *C* intercepts this message. After receiving the message, *C* uses K_{CS} to compute $f_{K_{CS}}(ID_A, ID_B, N_C)$ and verifies if the computation result is equal to the received one. If it is correct, *C* believes that he/she is communicating with a legitimate *S*; otherwise, *C* regards S illegal and terminates the protocol. Next, *C* uses K_{AC} to compute $f_{K_{AC}}(ID_A, ID_B, N_C)$ and verifies if the computation result is equal to the received one. If it is correct, *C* believes that he/she is communicating with a legitimate *A*; otherwise, *C* regards *A* illegal and terminates the protocol.

Finally, *A* and *C* can share the session key K_{AC} to encrypt and decrypt their communicated messages. *A* believes that it is communicating with B but actually it is communicating with *C*.

6. CONCLUSION

The key exchange protocol using passwords achieved great attention due to its simplicity and efficiency. Recently, Chang proposed a practical three-party key exchange (C-3 PEKE) protocol with round efficiency. Later, Lee and Chang presented an

off-line password guessing attack on C-3 PEKE protocol. In the present paper, an impersonation-of-the initiator attack and impersonation-of-the responder attack are demonstrated on C-3 PEKE protocol using the off-line password guessing attack proposed by Lee and Chang.

Fig 1: C-3 PEKE protocol

A **S** **B**

Choose R_A Choose R_{S_1}, R_{S_2} Choose R_B

Compute $N_A = g^{R_A} \bmod p$ Compute $N_{S_1} = g^{RS_1} \bmod p$ Compute $N_B = g^{R_B} \bmod p$

Compute $N_{S_2} = g^{RS_2} \bmod p$

$\xrightarrow{ID_A, ID_B, N_A}$

$$K_{AS} = N_A^{RS_1} \bmod p = g^{RS_1 R_A} \bmod p$$

$\xrightarrow{(ID_A, N_A, E_{3P_A}(N_{S_1}), E_{3P_B}(N_{S_2}), f_{K_{AS}}(ID_A, ID_B, N_A))}$

Decrypt $E_{3P_B}(N_{S_2})$, get N_{S_2}

Compute $K_{BS} = N_{S_2}{}^{R_B} \bmod p = g^{RS_2 R_B} \bmod p$

$K_{AB} = N_A{}^{R_B} \bmod p = g^{R_A R_B} \bmod p$

$\xleftarrow{ID_B, N_B, E_{3P_A}(N_{S_1}), f_{K_{AS}}(ID_A, ID_B, N_A), f_{K_{AB}}(ID_A, ID_B, N_A), f_{K_{BS}}(ID_A, ID_B, N_{S_2})}$

Decrypt $E_{3P_B}(N_{S_1})$, get N_{S_1}

$K_{AS} = N_{S_1}{}^{R_A} \bmod p = g^{RS_1 R_A} \bmod p$

$K_{AB} = N_B{}^{R_A} \bmod p = g^{R_A R_B} \bmod p$

verify $f_{K_{AS}}(ID_A, ID_B, N_A)$,

(A authenticates *S*)

verify $f_{K_{AB}}(ID_A, ID_B, N_A)$,

(*A* authenticates *B*)

$\xrightarrow{ID_A, ID_B, N_B, f_{K_{AS}}(ID_A, ID_B, N_{S_1}), f_{K_{BS}}(ID_A, ID_B, N_{S_2}), f_{K_{AB}}(ID_A, ID_B, N_B)}$

$K_{BS} = N_B{}^{RS_2} \bmod p = g^{RS_2R_B} \bmod p$

Verify $f_{K_{AS}}(ID_A, ID_B, N_{S_1})$,

(*S* authenticates *A*)

Verify $f_{K_{BS}}(ID_A, ID_B, N_{S_2})$,

(*S* authenticates *B*)

$\xrightarrow{ID_A, f_{K_{AB}}(ID_A, ID_B, N_B), f_{K_{BS}}(ID_A, ID_B, N_B)}$

Verify $f_{K_{BS}}(ID_A, ID_B, N_B)$

(*B* authenticates *S*)

Verify $f_{K_{AB}}(ID_A, ID_B, N_B)$

(*B* authenticates *A*)

K_{AB} is the final key through which *A* and *B* communicates

Fig 2: Impersonation of the initiator attack on C-3PEKE protocol

Attacker(C)	**S**	**B**
Choose R_C	Choose R_{S_1}, R_{S_2}	Choose R_B
Compute $N_C = g^{R_C} \bmod p$	Compute $N_{S_1} = g^{RS_1} \bmod p$	Compute $N_B = g^{R_B} \bmod p$
	Compute $N_{S_2} = g^{RS_2} \bmod p$	
$\xrightarrow{(ID_A, ID_B, N_C)}$		

$$K_{CS} = N_C{}^{RS_1} \bmod p = g^{RS_1R_C} \bmod p$$

$$\xrightarrow{(ID_A, N_C, E_{3P_A}(N_{S_1}), E_{3P_B}(N_{S_2}), f_{K_{CS}}(ID_A.ID_B, N_C))}$$

Decrypt $E_{3P_B}(N_{S_2})$, get N_{S_2}

Compute $K_{BS} = N_{S_2}{}^{R_B} \bmod p = g^{RS_2R_B} \bmod p$

$K_{CB} = N_C{}^{R_B} \bmod p = g^{R_CR_B} \bmod p$

$$\xleftarrow{ID_B, N_B, E_{3P_A}(N_{S_1}), f_{K_{CS}}(ID_A.ID_B, N_C), f_{K_{CB}}(ID_A.ID_B, N_C), f_{K_{BS}}(ID_A.ID_B, N_{S_2})}$$

(Since password of *A* i.e. P_A is already

determined as shown in section 2(A))

Decrypt $E_{3P_A}(N_{S_1})$, get N_{S_1}

$K_{CS} = N_{S_1}{}^{R_C} \bmod p = g^{RS_1R_C} \bmod p$

$K_{CB} = N_B{}^{R_C} \bmod p = g^{R_CR_B} \bmod p$

verify $f_{K_{CS}}(ID_A, ID_B, N_C)$,

(*A* authenticates *S*)

verify $f_{K_{CB}}(ID_A, ID_B, N_C)$,

(*C* authenticates *B*)

$$\xrightarrow{ID_A, ID_B, N_B, f_{K_{CS}}(ID_A, ID_B, N_{S_1}), f_{K_{BS}}(ID_A, ID_B, N_{S_2}), f_{K_{CB}}(ID_A, ID_B, N_B)}$$

$K_{BS} = N_B{}^{RS_2} \bmod p = g^{RS_2R_B} \bmod p$

Verify $f_{K_{CS}}(ID_A, ID_B, N_{S_1})$,

(*S* authenticates *A* (but it is actually *C*))

Verify $f_{K_{BS}}(ID_A, ID_B, N_{S_2})$,

(*S* authenticates *B*)

$$\xrightarrow{ID_A, f_{K_{CB}}(ID_A, ID_B, N_B), f_{K_{BS}}(ID_A, ID_B, N_B)}$$

Verify $f_{K_{BS}}(ID_A, ID_B, N_B)$

(*B* authenticates *S*)

Verify $f_{K_{CB}}(ID_A, ID_B, N_B)$

(*B* authenticates *A*)

K_{CB} is the final key trough which *C* and *B* communicates

B thinks that it is communicating with *A* but it is actually communicating with *C* (attacker).

Fig 3: Impersonation-of-responder attack on C-3PEKE protocol

A	**S**	**Attacker (C)**
Choose R_A	Choose R_{S_1}, R_{S_2}	Choose R_C
Compute $N_A = g^{R_A} \bmod p$	Compute $N_{S_1} = g^{RS_1} \bmod p$	Compute $N_C = g^{R_C} \bmod p$
	Compute $N_{S_2} = g^{RS_2} \bmod p$	
$\xrightarrow{ID_A, ID_B, N_A}$		
	$K_{AS} = N_A^{RS_1} \bmod p = g^{RS_1 R_A} \bmod p$	
	$\xrightarrow{ID_A, N_A, E_{3P_A}(N_{S_1}), E_{3P_B}(N_{S_2}), f_{K_{AS}}(ID_A.ID_B, N_A)}$	
		(*C* already obtained *B*'s password by mounting off-line password guessing attack as shown in section 2(*A*))
		Decrypt $E_{3P_B}(N_{S_2})$, get N_{S_2}
		Compute $K_{CS} = N_{S_2}^{R_C} \bmod p = g^{RS_2 R_C} \bmod p$
		$K_{AC} = N_A^{R_C} \bmod p = g^{R_A R_C} \bmod p$
	$\xleftarrow{ID_B, N_C, E_{3P_A}(N_{S_1}), f_{K_{AS}}(ID_A.ID_B, N_A), f_{K_{AC}}(ID_A.ID_B, N_A), f_{K_{CS}}(ID_A.ID_B, N_{S_2})}$	
Decrypt $E_{3P_A}(N_{S_1})$, get N_{S_1}		

$K_{AS} = N_{S_1}{}^{R_A} \bmod p = g^{RS_2R_A} \bmod p$

$K_{AC} = N_C{}^{R_A} \bmod p = g^{R_AR_C} \bmod p$

verify $f_{K_{AS}}(ID_A, ID_B, N_A)$,

(*A* authenticates *S*)

verify $f_{K_{AC}}(ID_A, ID_B, N_A)$,

(*A* authenticates *B*)

(But it is actually *C*)

$$\xrightarrow{ID_A, ID_B, N_C, f_{K_{AS}}(ID_A, ID_B, N_{S_1}), f_{K_{CS}}(ID_A, ID_B, N_{S_2}), f_{K_{AC}}(ID_A, ID_B, N_C)}$$

$K_{CS} = N_C{}^{RS_2} \bmod p = g^{RS_2R_C} \bmod p$

Verify $f_{K_{AS}}(ID_A, ID_B, N_{S_1})$

(*S* authenticates *A*)

Verify $f_{K_{CS}}(ID_A, ID_B, N_{S_2})$,

(*S* authenticates *B*)

(But it is actually *C*)

$$\xrightarrow{ID_A, f_{K_{AC}}(ID_A, ID_B, N_B), f_{K_{CS}}(ID_A, ID_B, N_B)}$$

Verify $f_{K_{CS}}(ID_A, ID_B, N_C)$

(*C* authenticates *S*)

Verify $f_{K_{AC}}(ID_A, ID_B, N_C)$

(*C* authenticates *A*)

K_{AC} is the final key through which *A* and *B* communicates

7. REFERENCES

[1] Abdalla, M., Chevassut, O., and Pointcheval, D. *One-time verifier-based encrypted key exchange*, Proc. of PKC '05, LNCS 3386, Springer-Verlag, pp. 47–64, 2005.

[2] Abdalla, M., and Pointcheval, D. *Simple Password-Based Encrypted Key Exchange Protocols*, Proc. of Topics in Cryptology - CT-RSA, LNCS 3376, Springer-Verlag, pp. 191-208, 2005.

[3] Abdalla, M., and Pointcheval, D. *Interactive Diffie-Hellman Assumptions with Applications to Password-based Authentication*, Proceedings of the 9th International Conference on Financial Cryptography (FC'2005), Roseau, Dominica, Berlin, Germany: Springer-Verlag, pp.341-356, 2005.

[4] Abdalla, M., Fouque, P, A., Pointcheval, D. *Password-based authenticated key exchange in the three-party setting*, Proceedings of the 8th International Workshop on Theory and Practice in Public Key Cryptography (PKC'2005). Berlin, Germany: Springer-Verlag, 2005:65-84. Full version appeared in IEE Information Security, v 153(1), pp. 27–39, March 2006.

[5] Bellovin, S M., and Merritt, M. *Encrypted key exchange: Password-based protocols secure against dictionary*

attacks, Proc. 1992 IEEE Symposium on Security and Privacy, pp. 72-84, May 1992.

[6] Bellare, M., Pointcheval, D., and Rogaway, P. *Authenticated key exchange secure against dictionary attacks*, Proceedings of the 2000 Advances in Cryptology (EUROCRYPT'2000). Berlin, Germany: Springer-Verlag, pp. 139-155, 2000.

[7] Bresson, E., Chevassut, O., and Pointcheval, D. *New security results on encrypted key exchange*, Proc. PKC 2004, LNCS 2947, Springer-Verlag, pp. 145-158. Mar. 2004.

[8] Chang, Y, F. *A practical Three-party key exchange protocol with Round Efficiency*, International Journal of Innovative Computing, Information and control, v 4(4), pp. 953-960, April 2008.

[9] Ding. Y, Horster, P. *Undetectable on-line password guessing attacks*, ACM Operat Syst Rev 29(4), pp.77– 86, 1995.

[10] Diffie, W., and Hellman, M. *New Directions in cryptography* , IEEE Transactions on Information theory, v 22 (6) , pp. 644-654, 1976.

[11] Kim and Choi. *Enhanced Password-based simple three-party Key exchange protocol*, Computers and Electrical Engineering, v 35(1), pp107-114, 2009.

[12] K, Kobara., and H, Imai. *Pretty-simple password-authenticated key exchange under standard assumptions*, IEICE Transactions, E85-A (10):pp.2229-2237, Oct. 2002. Also available at http://eprint.iacr.org/2003/038/.

[13] Lee., and Chang, *On security of a three party key exchange protocol with round efficiency*, Information technology and control, kaunas, technologija, v 37(4), pp.333-335, 2008.

[14] Lee, T, F., Hwang, T., and Lin, C, L. *Enhanced three-party encrypted key exchange without server's public keys*, Computers and Security, 23(7): pp.571-577, 2004.

[15] Lee, S, W., Kim, H, S., and Yoo, K, Y. *Efficient verifier-based key agreement for three parties without server's public key*, Applied Mathematics and Computation, 167(2), pp. 996-1003, 2005.

[16] Lin, C, L., Sun, H, M., Steiner, M., and Hwang T. *Three-party encrypted key exchange without server's public keys*, IEEE Communications Letters, v5(12), pp. 497-499, 2001.

[17] Lin, C, L., Sun, H, M and Hwang T. *Three-party encrypted key exchange attacks and a solution*, ACM Operating Systems Review, 34(4), pp.12-20, 2000.

[18] Lu, R., and Cao, Z. *Simple three-party key exchange protocol*, Computers and Security, v 26(1), pp.94-97, 2007.

[19] Steiner, M., Tsudik, G., and Waidner, M. *Refinement and extention of encrypted key exchange*, ACM Operating Systems Review, v 29(3), pp 22-30, 1995.