IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Volume-3, Issue-12 E-ISSN: 2347-2693

Probe of Password Authenticated 3P-EKE Protocols

Archana Raghuvamshi[1]* and Premchand Parvataneni[2]

[1*]Department of Computer Science & Engineering, Adkavi Nannaya University, Rajahmundry, India
[2] Department of Computer Science & Engineering, Osmania University, Hyderabad, India

www.ijcseonline.org

Received: 26/Nov/2015 Revised: 09/Dec/2015 Accepted: 20/Dec/2015 Published: 31/Dec/2015

***Abstract*—** To establish a secure communication over unreliable networks, Password-Authenticated Encrypted Key Exchange (PA-EKE) protocols plays a pivotal role. Diffie & Hellman (D-H) (1976) proposed a first key agreement protocol, which is suffered from man-in-the-middle attack. To overcome a flaw, many authors proposed password-authenticated key agreement protocols. Chang & Chang (2004) proposed a novel password-authenticated 3P-EKE protocol with round efficiency. In contrast, Yoon & Yoo (2008) notified an undetectable online dictionary attack on this protocol and proposed an improvement over it. Later, Padmavathy et al. cryptanalyzed the Yoon-Yoo's protocol and proposed PSRJ Protocol. Subsequently, Archana et al. (2012) notified a detectable online dictionary attack on PSRJ Protocol. Afterward, an improvement over the Yoon-Yoo's protocol is proposed by Raj et al. (2013), which is cryptanalyzed by Archana et al. (2013). In this paper we have analyzed all the above protocols at their performance level.

***Keywords* --** Password authentication, Encrypted Key Exchange (EKE), Three Party-EKE Protocols, , Detectable On-line Dictionary Attacks, Undetectable On-line Dictionary Attacks.

I. INTRODUCTION

Within this new era of unreliable network to form a secure communication is a challenging task. Many password-based authenticated encrypted key exchange protocols (PAEKE) is widely setup due its ease and easy to maintaining human memorable passwords. To establish a secured session key each interacting party shares a low entropy password with the trusted third-party. Such kind of protocols in which trusted third-party is involved in establishing a secured communication is known as 3P-EKE (three-party encrypted key exchange) protocols. There are four *Requirements* for the implementation of such protocols (3P-EKE); Efficiency, Practicability, Mutual Authentication, Confidence and decisiveness [1].In general, 2P-EKE and 3P-EKE protocols may suffer from any one of the following three types of dictionary attacks according to Ding & Horster [2].

1. Detectable on-line dictionary attacks.
2. Undetectable on-line dictionary attacks.
3. Off-line dictionary attacks.

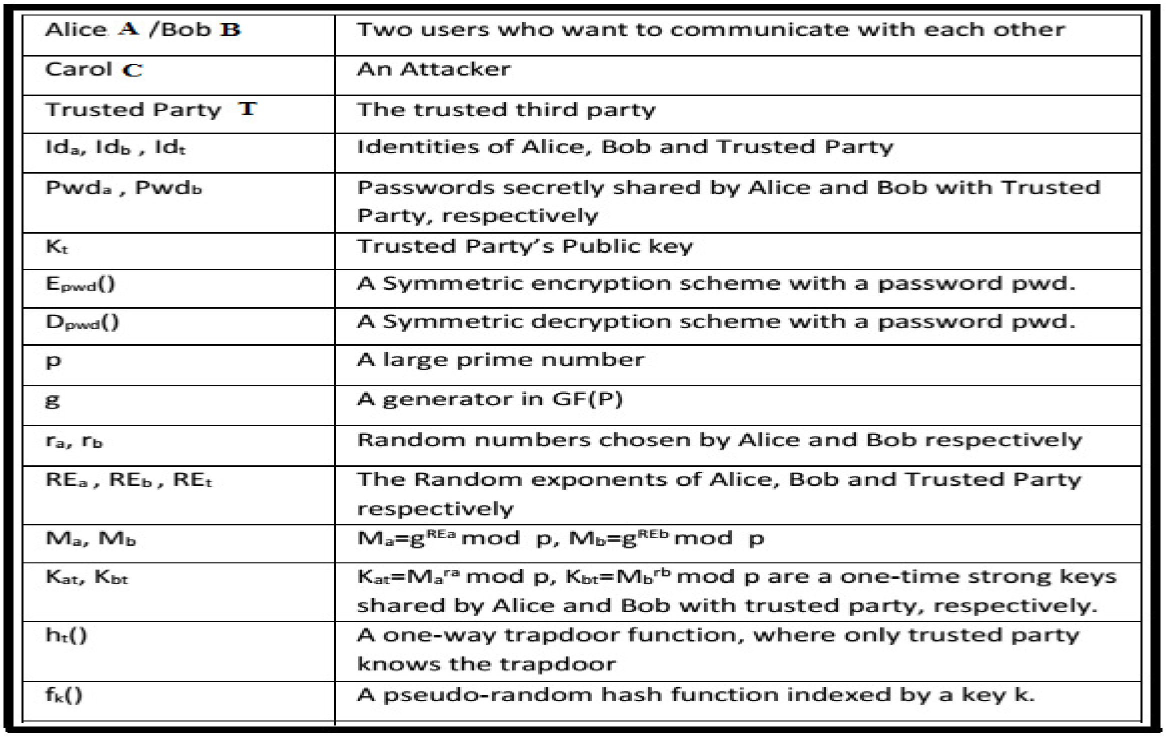
There are four *Necessities* for the implementation of such 3P-EKE protocols [1]; Efficiency, Practicability, Mutual Authentication, Confidence and decisiveness.

i. *Efficiency:* For determining the efficiency of a protocol, the round efficiency and computation complexity are all taken into consideration.
ii. *Practicability:* no credential is needed.
iii. *Mutual authentication:* The clients of a protocol including server has to mutually authenticate each other. For e.g., Alice A, Bob B, and Trusted Party T can authenticate with one another.
iv. *Confidence and decisiveness*: a protocol can defend against three classes of attacks iff easy-to-remember passwords involved satisfies confidence and decisiveness.

Diffie-Hellman [3] has proposed a Key Agreement Protocol in 1976, which is suffered from man-in-the-middle attack. Later, to overcome the flaw many authors proposed many key agreement protocols. In 1992, Bellovin and Merrit [4] proposed first password-based 2P-EKE (Two-Party Encrypted key exchange) protocol. Later, many two-party Password-based Authenticated Encrypted Key Exchange (2PAEKE) Protocols have been probed. Each pair of interacting party has to share individual password for each session in 2P-EKE protocol (user-user model), which raise a difficulty of maintaining n! (Exponential) Passwords. This inadequacy enthuses research community to incorporate 2PAEKE protocols into 3PAEKE schemes, i.e. user-server-user model by Steiner et al. [5] (STW-3PAKE Protocol).Ding & Horster [2] shown an undetectable on-line password guessing attacks on this STW-3PAKE Protocol.
In 2000, Lin et al. [6] proposed an off-line password guessing attacks on this STW-3PAKE Protocol and also given an improvement for it by using public key cryptosystem. In 2002, Zhu et al. proposed a password

authenticated key exchange protocol based on RSA [7]. Later in 2003, Yeh et al. has shown that Zhu et al.'s protocol suffers from the undetectable password-guessing attacks and also has given solutions for improvement[8]. In 2004, Chang and Chang proposed a novel three party simple key exchange protocol [1].In the same year, due to the heavy computation of public key in Lin et al. protocol it is proved to be inefficient. Hence, Lee et al. [9] proposed a new enhanced protocol by sacrificing heavy computation of public key.

Table 1.List of Notations



Alice A /Bob B	Two users who want to communicate with each other
Carol C	An Attacker
Trusted Party T	The trusted third party
Id_a, Id_b , Id_t	Identities of Alice, Bob and Trusted Party
Pwd_a , Pwd_b	Passwords secretly shared by Alice and Bob with Trusted Party, respectively
K_t	Trusted Party's Public key
$E_{pwd}()$	A Symmetric encryption scheme with a password pwd.
$D_{pwd}()$	A Symmetric decryption scheme with a password pwd.
p	A large prime number
g	A generator in GF(P)
r_a, r_b	Random numbers chosen by Alice and Bob respectively
RE_a , RE_b , RE_t	The Random exponents of Alice, Bob and Trusted Party respectively
M_a, M_b	$M_a=g^{REa}$ mod p, $M_b=g^{REb}$ mod p
K_{at}, K_{bt}	$K_{at}=M_a^{ra}$ mod p, $K_{bt}=M_b^{rb}$ mod p are a one-time strong keys shared by Alice and Bob with trusted party, respectively.
$h_t()$	A one-way trapdoor function, where only trusted party knows the trapdoor
$f_k()$	A pseudo-random hash function indexed by a key k.

In 2008, an undetectable online dictionary attack has notified on chang-chang's protocol and further improvement has been proposed by Yoon-Yoo[10]. Later in 2010, Padmavathy et al. [11] cryptanalyzed the Yoon-Yoo Protocol and proposed an improved 3P-EKE (PSRJ Protocol) Protocol by proving the proposed protocol sued to be strong to undetectable online dictionary attack. In 2012, Archana et al. [12] publicized a detectable online dictionary attack on PSRJ Protocol. An improvement on the Yoon-Yoo's Protocol has been proposed by Raj et al.[13] which is proved to be vulnerable to detectable online dictionary attack by Archana et al. [14] in 2013.

In Section II, we listed notations used in the various protocols throughout this paper. Section III, describes the analysis of Chang-Chang's protocol. In Section IV, we have probed Yoon-Yoo's Protocol. In Section V, we reviewed PSRJ Protocol and its weakness. Section VI, describes the analysis of Raj et al. Protocol. Performance Analysis of these protocols is done in Section VII. Final remarks are made in section VIII.

II. NOTATIONS

The notations used throughout this paper are listed in Table 1. The protocols discussed in this paper assume that the passwords Pwd_a (Alice) and Pwd_b (Bob) are initially shared with a trusted party through a secured channel.

III. PROBE OF CHANG-CHANG'S PROTOCOL

This section devotes to review the Chang-Chang's protocol and then how it is cryptanalyzed by Yoon & Yoo i.e., how it is defenseless to undetectable on-line dictionary attack is shown.

A. *Review*

The detailed procedure of the Chang-Chang's protocol is described in different steps as follows:

Step 1: Alice selects two random numbers r_a, $RE_a \in_R Z_p$ and computes $M_a=g^{REa}$(mod p) & $K_{at}=M_a^{ra}$ (mod p). Now, Alice uses her password Pwd_a to encrypt M_a and also computes $h_t(r_a)$ & $f_{Kat}(M_a)$. Then she transfers {id_a, id_b, id_t, $E_{Pwda}(M_a)$,$h_t(r_a)$, $f_{Kat}(M_a)$} to Bob.

i.e., A→B: {id_a, id_b, id_t, $E_{Pwda}(M_a)$,$h_t(r_a)$, $f_{Kat}(M_a)$}

Step 2: After getting the credentials from Alice, Bob selects two random numbers r_b,$RE_b \in_R Z_p$ and computes

$M_b=g^{REb}$(mod p) & $K_{bt}=M_b^{rb}$(mod p). Now, Bob uses his password Pwd_b to encrypt M_b and also computes $h_t(r_b)$ & $f_{Kbt}(M_b)$. Then he transfers {id_a, id_b, id_t, $E_{Pwda}(M_a)$,$h_t(r_a)$, $f_{Kat}(M_a)$, $E_{Pwdb}(M_b)$,$h_t(r_b)$, $f_{Kbt}(M_b)$} to Trusted Party.

i.e., B→T: {id_a, id_b, id_t, $E_{Pwda}(M_a)$, $h_t(r_a)$, $f_{Kat}(M_a)$, $E_{Pwdb}(M_b)$, $h_t(r_b)$, $f_{Kbt}(M_b)$}

i.e., B→A: {M_b^{REt}mod p, $f_{Kat}(id_a$, id_b, K_{at}, M_b^{REt}mod p), $f_K(id_b$, K)}

Step 5: Alice authenticates Trusted Party by checking $f_{Kat}(id_a$, id_b, K_{at}, M_b^{REt}mod p). If successful, Alice authenticates Bob by checking $f_K(id_b$, K) and computes the session key K=(M_b^{REt}) ^{REa} mod p=(($g_b^{REb})^{REt})^{REa}$ mod p and sends $f_K(id_a$, K) to Bob.

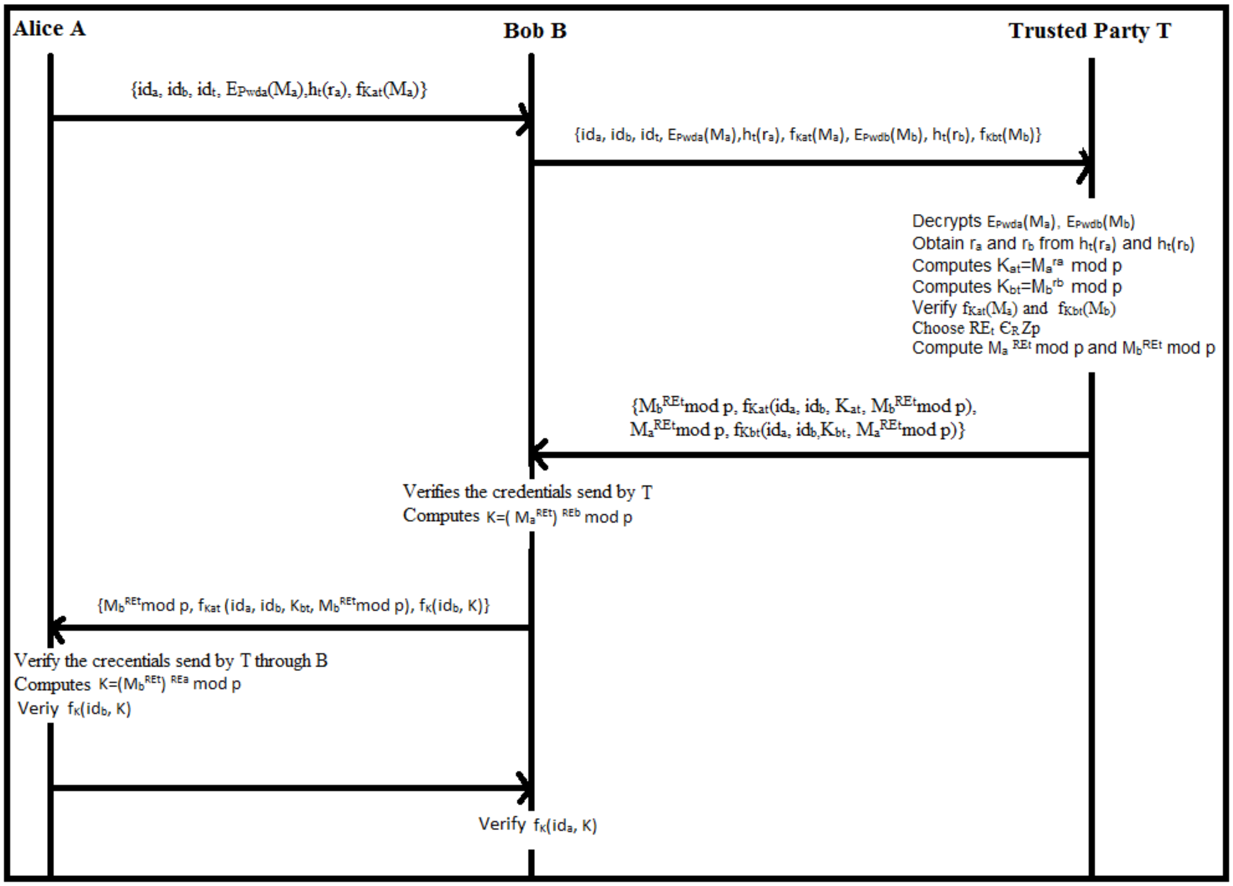

Alice A
Bob B
Trusted Party T
{id_a, id_b, id_t, $E_{Pwda}(M_a)$,$h_t(r_a)$, $f_{Kat}(M_a)$}
{id_a, id_b, id_t, $E_{Pwda}(M_a)$,$h_t(r_a)$, $f_{Kat}(M_a)$, $E_{Pwdb}(M_b)$, $h_t(r_b)$, $f_{Kbt}(M_b)$}
Decrypts $E_{Pwda}(M_a)$, $E_{Pwdb}(M_b)$
Obtain r_a and r_b from $h_t(r_a)$ and $h_t(r_b)$
Computes $K_{at}=M_a^{ra}$ mod p
Computes $K_{bt}=M_b^{rb}$ mod p
Verify $f_{Kat}(M_a)$ and $f_{Kbt}(M_b)$
Choose $RE_t \in_R Zp$
Compute M_a^{REt} mod p and M_b^{REt} mod p
{M_b^{REt}mod p, $f_{Kat}(id_a$, id_b, K_{at}, M_b^{REt}mod p),
M_a^{REt}mod p, $f_{Kbt}(id_a$, id_b,K_{bt}, M_a^{REt}mod p)}
Verifies the credentials send by T
Computes K=(M_a^{REt}) ^{REb} mod p
{M_b^{REt}mod p, f_{Kat} (id_a, id_b, K_{bt}, M_b^{REt}mod p), $f_K(id_b$, K)}
Verify the crecentials send by T through B
Computes K=(M_b^{REt}) ^{REa} mod p
Veriy $f_K(id_b$, K)
Verify $f_K(id_a$, K)

Fig 1: Chang-Chang's 3P-EKE Protocol

Step 3: Trusted Party uses Pwd_a/Pwd_b and a trapdoor [15] t to get M_a/M_b and r_a/r_b to compute K_{at}/K_{bt} to authenticate Alice/Bob by checking $f_{Kat}(M_a)/f_{Kbt}(M_b)$.If successful, Trusted Party computes the credentials {M_b^{REt} mod p, $f_{Kat}(id_a$, id_b, K_{at}, M_b^{REt}mod p), M_a^{REt}mod p, $f_{Kbt}(id_a$, id_b, K_{bt}, M_a^{REt} mod p)} and sends it to Bob.

i.e., T→B: {M_b^{REt}mod p, $f_{Kat}(id_a$, id_b, K_{at}, M_b^{REt} mod p), M_a^{REt}mod p, $f_{Kbt}(id_a$, id_b,K_{bt}, M_a^{REt}mod p)}

Step 4: Bob authenticates Trusted Party by checking $f_{Kbt}(id_a$, id_b, K_{bt}, M_a^{REt}mod p). If successful, Bob computes the session key K=($M_a^{REt})^{REb}$ mod p =(($g_a^{REa})^{REt})^{REb}$mod p and sends the credentials {M_b^{REt}mod p, $f_{Kat}(id_a$, id_b, K_{at}, M_b^{REt}mod p), $f_K(id_b$, K)} to Alice.

i.e.,A→B: $f_K(id_a$, K)

Step 6: Bob verifies $f_K(id_a$, K) to authenticate Alice. If it is valid, Both Alice and Bob can communicate securely by using a common session key K.

The detail explanation of protocol is depicted in Fig 1.

B. Attack

Yoon & Yoo are notified an undetectable online dictionary attack on Chang-Chang's protocol by assuming Bob as malevolent party. The Procedure of attack is given below:

Step 1: Alice sends $\{id_a, id_b, id_t, E_{Pwda}(M_a),h_t(r_a), f_{Kat}(M_a)\}$ to Bob
i.e., A →B: $\{id_a, id_b, id_t, E_{Pwda}(M_a),h_t(r_a), f_{Kat}(M_a)\}$

Step 2: Bob records the message $\{id_a, id_b, id_t, E_{Pwda}(M_a),h_t(r_a), f_{Kat}(M_a)\}$ sent from Alice.

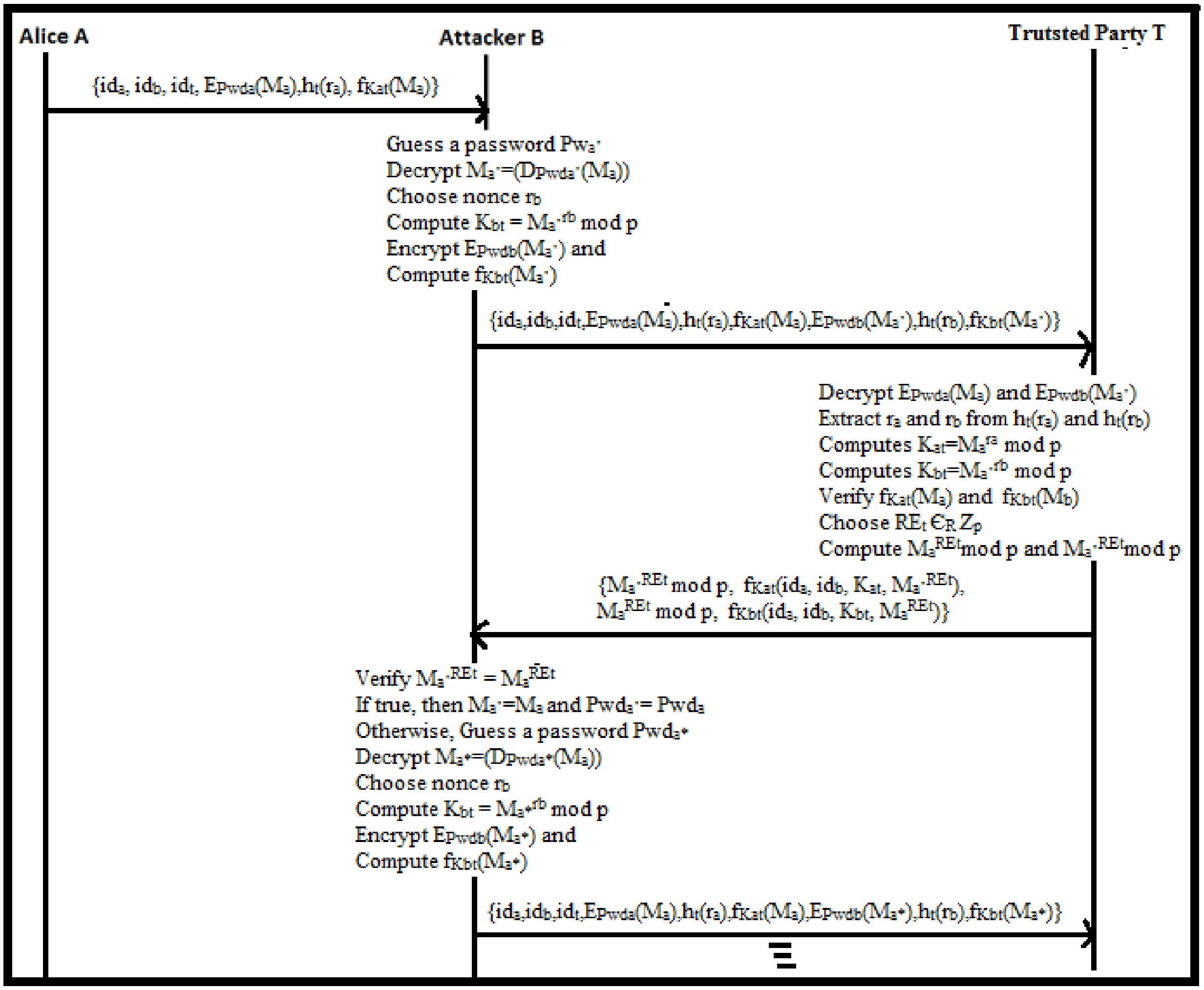

Alice A
Attacker B
Trutsted Party T
{ida, idb, idt, EPwda(Ma),ht(ra), fKat(Ma)}
Guess a password Pwa'
Decrypt Ma'=(DPwda'(Ma))
Choose nonce rb
Compute Kbt = Ma'^rb mod p
Encrypt EPwdb(Ma') and
Compute fKbt(Ma')
{ida,idb,idt,EPwda(Ma),ht(ra),fKat(Ma),EPwdb(Ma'),ht(rb),fKbt(Ma')}
Decrypt EPwda(Ma) and EPwdb(Ma')
Extract ra and rb from ht(ra) and ht(rb)
Computes Kat=Ma^ra mod p
Computes Kbt=Ma'^rb mod p
Verify fKat(Ma) and fKbt(Mb)
Choose REt ∈R Zp
Compute Ma^REt mod p and Ma'^REt mod p
{Ma'^REt mod p, fKat(ida, idb, Kat, Ma'^REt),
Ma^REt mod p, fKbt(ida, idb, Kbt, Ma^REt)}
Verify Ma'^REt = Ma^REt
If true, then Ma'=Ma and Pwda'= Pwda
Otherwise, Guess a password Pwda*
Decrypt Ma*=(DPwda*(Ma))
Choose nonce rb
Compute Kbt = Ma*^rb mod p
Encrypt EPwdb(Ma*) and
Compute fKbt(Ma*)
{ida,idb,idt,EPwda(Ma),ht(ra),fKat(Ma),EPwdb(Ma*),ht(rb),fKbt(Ma*)}

Fig 2: Undetectable Online Dictionary Attack on Chang-Chang's Protocol

Step 3: Bob guess Alice's password as $Pwd_{a'}$ and by decrypting i.e., $D_{Pwda'}(E_{Pwda}(M_a))$ gets $M_{a'}$.
Step 4: Bob selects a random number $r_b \in_R Z_p$ to compute $K_{bt} = M_{a'}^{rb}$ mod p and encrypts $M_{a'}$ by using his password Pwd_b. Finally, Bob computes two hash values $h_t(r_b)$ and $f_{Kbt}(M_{a'})$ and
transmits $\{id_a, id_b, id_t, E_{Pwda}(M_a), h_t(r_a), f_{Kat}(M_a), E_{Pwdb}(M_{a'}), h_t(r_b), f_{Kbt}(M_{a'})\}$ to T.
i.e., B →T: $\{ id_a, id_b, id_t, E_{Pwda}(M_a),h_t(r_a), f_{Kat}(M_a), E_{Pwdb}(M_{a'}), h_t(r_b), f_{Kbt}(M_{a'})\}$

Step 5: Trusted Party uses Pwd_a/Pwd_b & a trapdoor t [15] to get $M_a/M_{a'}$ & r_a/r_b and computes K_{at}/K_{bt} to authenticate Alice /Bob by checking $f_{Kat}(M_a)/f_{Kbt}(M_{a'})$.If successful, Trusted Party computes the credentials $\{M_{a'}^{REt}$mod p, $f_{Kat}(id_a, id_b, K_{at}, M_{a'}^{REt}$ mod p), M_a^{REt}mod p, $f_{Kbt}(id_a, id_b, K_{bt}, M_a^{REt}$mod p$)\}$ and sends it to Bob.
i.e., T →B: $\{M_{a'}^{REt}$ mod p, $f_{Kat}(id_a, id_b, K_{at}, M_{a'}^{REt})$, M_a^{REt} mod p, $f_{Kbt}(id_a, id_b, K_{bt}, M_a^{REt})\}$
Step 6: After receiving the message, Bob simply compares $M_{a'}^{REt} = M_a^{REt}$. If $M_{a'}^{REt} = M_a^{REt}$ then it follows that $Pwd_{a'}= Pwd_a$. Hence succeed.

However, the attack cannot be detected by Trusted Party. As a result, undetectable on-line dictionary attacks can be easily mounted on Chang-Chang's protocol. The procedure for attack also illustrated in Fig 2.

IV. PROBE OF YOON-YOO'S PROTOCOL

Yoon and Yoo proposed an improvement over the Chang-Chang's 3P-EKE scheme. They claimed that the proposed protocol can defend against undetectable on-line password guessing attacks. This section devotes to first review the Yoon-Yoo's protocol and then how it is cryptanalyzed by Padmavathy et al. i.e., how it is susceptible to undetectable on-line password guessing attack has shown.

A. *Review*

The Yoon and Yoo's 3P-EKE protocol is demonstrated in Fig 3.The detail of Yoon-Yoo's protocol is described as follows:

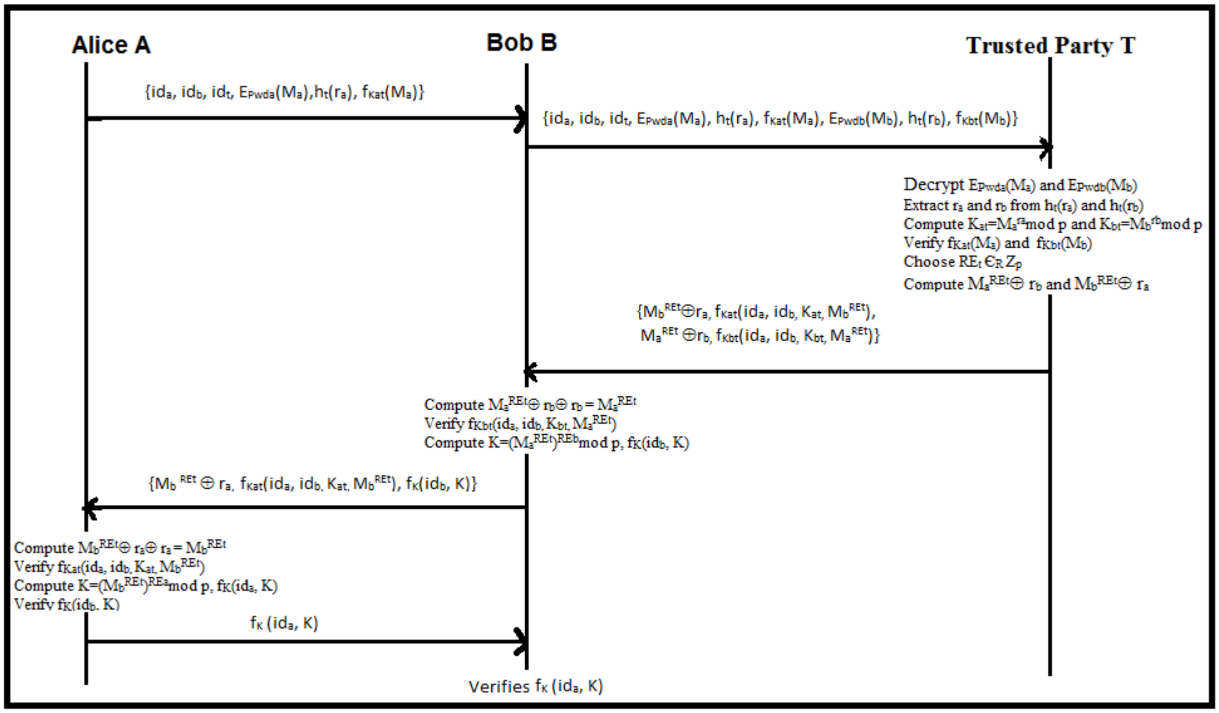

Alice A
Bob B
Trusted Party T
$\{id_a, id_b, id_t, E_{Pwda}(M_a), h_t(r_a), f_{Kat}(M_a)\}$
$\{id_a, id_b, id_t, E_{Pwda}(M_a), h_t(r_a), f_{Kat}(M_a), E_{Pwdb}(M_b), h_t(r_b), f_{Kbt}(M_b)\}$
Decrypt $E_{Pwda}(M_a)$ and $E_{Pwdb}(M_b)$
Extract r_a and r_b from $h_t(r_a)$ and $h_t(r_b)$
Compute $K_{at}=M_a^{ra}$mod p and $K_{bt}=M_b^{rb}$mod p
Verify $f_{Kat}(M_a)$ and $f_{Kbt}(M_b)$
Choose $RE_t \in_R Z_p$
Compute $M_a^{REt}\oplus r_b$ and $M_b^{REt}\oplus r_a$
$\{M_b^{REt}\oplus r_a, f_{Kat}(id_a, id_b, K_{at}, M_b^{REt}),$
$M_a^{REt}\oplus r_b, f_{Kbt}(id_a, id_b, K_{bt}, M_a^{REt})\}$
Compute $M_a^{REt}\oplus r_b\oplus r_b = M_a^{REt}$
Verify $f_{Kbt}(id_a, id_b, K_{bt}, M_a^{REt})$
Compute $K=(M_a^{REt})^{REb}$mod p, $f_K(id_b, K)$
$\{M_b^{REt}\oplus r_a, f_{Kat}(id_a, id_b, K_{at}, M_b^{REt}), f_K(id_b, K)\}$
Compute $M_b^{REt}\oplus r_a\oplus r_a = M_b^{REt}$
Verify $f_{Kat}(id_a, id_b, K_{at}, M_b^{REt})$
Compute $K=(M_b^{REt})^{REa}$mod p, $f_K(id_a, K)$
Verify $f_K(id_b, K)$
$f_K(id_a, K)$
Verifies $f_K(id_a, K)$

Fig 3: Yoon-Yoo's Password Authenticated 3P-EKE Protocol

Step 1: Alice selects two random numbers $r_a, RE_a \in_R Z_p$ to compute $M_a=g^{REa}$(mod p) and then $K_{at}=M_a^{ra}$ (mod p). Now Alice uses her password Pwd_a to encrypt M_a and computes $h_t(r_a)$ and $f_{Kat}(M_a)$. Then she transfers $\{id_a, id_b, id_t, E_{Pwda}(M_a), h_t(r_a), f_{Kat}(M_a)\}$ to Bob.
i.e., A→B: $\{id_a, id_b, id_t, E_{Pwda}(M_a), h_t(r_a), f_{Kat}(M_a)\}$

Step 2: Upon getting the credentials from Alice, Bob selects two random numbers $r_b, RE_b \in_R Z_p$ to compute $M_b=g^{REb}$(mod p) and then $K_{bt}=M_b^{rb}$(mod p). Now Bob takes his password Pwd_b to encrypt M_b and computes $h_t(r_b)$ and $f_{Kbt}(M_b)$. Then he transfers $\{id_a, id_b, id_s, E_{Pwda}(M_a), h_t(r_a), f_{Kat}(M_a), E_{Pwdb}(M_b), h_t(r_b), f_{Kbt}(M_b)\}$ to Trusted Party.
i.e., B→T: $\{id_a, id_b, id_t, E_{Pwda}(M_a), h_t(r_a), f_{Kat}(M_a), E_{Pwdb}(M_b), h_t(r_b), f_{Kbt}(M_b)\}$

Step 3: A Trusted Party uses the password of Alice (Pwd_a) and Bob (Pwd_b) to decrypt $E_{Pwda}(M_a)$ and $E_{Pwdb}(M_b)$ to get M_a and M_b. Then, by using trapdoor t a Trusted Party retrieves r_a & r_b from $h_t(r_a)$ & $h_t(r_b)$ and then computes $K_{at}=M_a^{ra}$(mod p) & $K_{bt}=M_b^{rb}$(mod p).Now, a Trusted Party authenticates to Alice and Bob by comparing the received $f_{Kat}(M_a)$ & $f_{Kbt}(M_b)$ with the computed $f_{Kat}(M_a)$ & $f_{Kbt}(M_b)$. If both are equal, then Trusted Party selects a random number $RE_t \in_R Z_p$ to compute M_a^{REt}(mod p) & M_b^{REt}(mod p) and then calculate $M_a^{REt}\oplus r_b$ and $M_b^{REs}\oplus r_a$. Now, Trusted Party computes $f_{Kat}(id_a, id_b, K_{at}, M_b^{REt})$ & $f_{Kbt}(id_a, id_b, K_{bt}, M_a^{REt})$ and sends the credentials $\{M_b^{REt}\oplus r_a, f_{Kat}(id_a, id_b, K_{at}, M_b^{REt}), M_a^{REt}\oplus r_b, f_{Kbt}(id_a, id_b, K_{bt}, M_a^{REt})\}$ to Bob.
i.e., T→B: $\{M_b^{REt}\oplus r_a, f_{Kat}(id_a, id_b, K_{at}, M_b^{REt}), M_a^{REt}\oplus r_b, f_{Kbt}(id_a, id_b, K_{bt}, M_a^{REt})\}$

Step 4: Bob uses r_b to compute $M_a^{REt}\oplus r_b\oplus r_b = M_a^{REt}$ and verifies $f_{Kbt}(id_a, id_b, K_{bt}, M_a^{REt})\}$. If it holds, Bob calculate a session key $K=(M_a^{REt})^{REb}$ mod p and then $f_K(id_b, K)$. Now, he sends the credentials $\{M_b^{REt}\oplus r_a, f_{Kat}(id_a, id_b, K_{at}, M_b^{REt}), f_K(id_b, K)\}$ to Alice.
i.e., B→ A: $\{M_b^{REt}\oplus r_a, f_{Kat}(id_a, id_b, K_{at}, M_b^{REt}), f_K(id_b, K)\}$

Step 5: After receiving the credentials from Bob, Alice uses r_a to compute $M_b^{REt}\oplus r_a\oplus r_a = M_b^{REt}$ and verifies $f_{Kat}(id_a, id_b, K_{at}, M_b^{REt})\}$. If it holds, Alice calculates a session key K=(

M_b^{REt}) ^{REa} mod p, f_K (id_b, K) and checks whether the computed result is equal to the received one. If it is equal, then Alice successfully validates Bob. Now, Alice computes f_K (id_a, K) and send it to Bob.
i.e., A→B: f_K (id_a, K)

Step 6: Bob verifies by computing $f_K(id_a, K)$ with the received f_K (id_a, K) to authenticate Alice. If it is valid, Both Alice and Bob can communicate by using a shared session key K.

B. Attack

This section exhibits the undetectable online dictionary attack on Yoon-Yoo's 3P-EKE protocol by Padmavathy et al. The attack is implemented by assuming that an unauthorized user Bob guess the Alice's password as $Pwd_{a'}$. This attack is illustrated in Fig 4.The details of this attack are as follows.

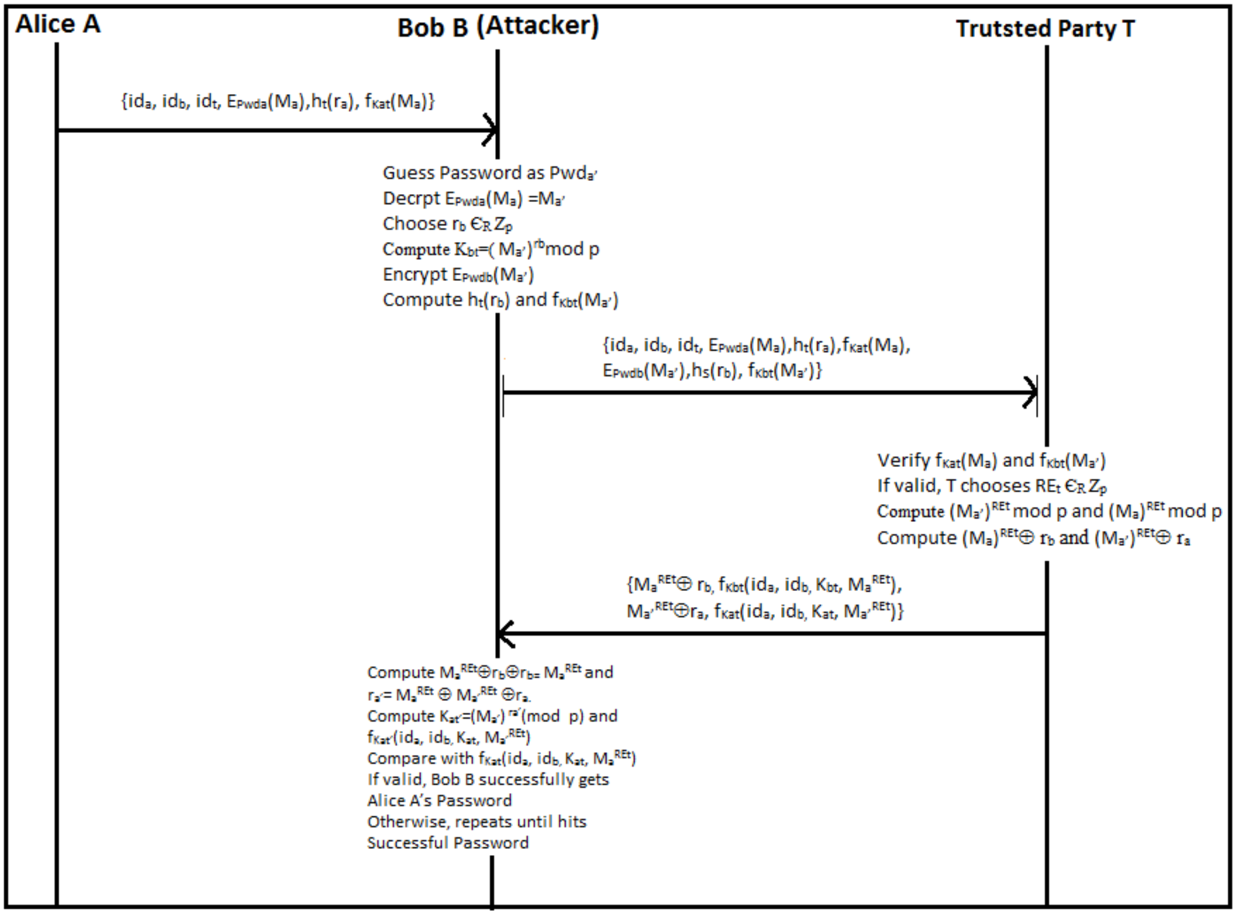

Alice A
Bob B (Attacker)
Trutsted Party T
{id_a, id_b, id_t, $E_{Pwda}(M_a)$,$h_t(r_a)$, $f_{Kat}(M_a)$}
Guess Password as $Pwd_{a'}$
Decrpt $E_{Pwda}(M_a)$ =$M_{a'}$
Choose $r_b \in_R Z_p$
Compute K_{bt}=($M_{a'}$)^{rb}mod p
Encrypt $E_{Pwdb}(M_{a'})$
Compute $h_t(r_b)$ and $f_{Kbt}(M_{a'})$
{id_a, id_b, id_t, $E_{Pwda}(M_a)$,$h_t(r_a)$,$f_{Kat}(M_a)$,
$E_{Pwdb}(M_{a'})$,$h_S(r_b)$, $f_{Kbt}(M_{a'})$}
Verify $f_{Kat}(M_a)$ and $f_{Kbt}(M_{a'})$
If valid, T chooses $RE_t \in_R Z_p$
Compute $(M_{a'})^{REt}$ mod p and $(M_a)^{REt}$ mod p
Compute $(M_a)^{REt}\oplus r_b$ and $(M_{a'})^{REt}\oplus r_a$
{$M_a^{REt}\oplus$ r_b, $f_{Kbt}(id_a, id_b, K_{bt}, M_a^{REt})$,
$M_{a'}^{REt}\oplus r_a$, $f_{Kat}(id_a, id_b, K_{at}, M_{a'}^{REt})$}
Compute $M_a^{REt}\oplus r_b \oplus r_{b=}$ M_a^{REt} and
$r_{a'}$= M_a^{REt} ⊕ $M_{a'}^{REt}$ ⊕r_a.
Compute $K_{at'}$=$(M_{a'})^{ra'}$(mod p) and
$f_{Kat'}(id_a, id_b, K_{at}, M_{a'}^{REt})$
Compare with $f_{Kat}(id_a, id_b, K_{at}, M_a^{REt})$
If valid, Bob B successfully gets
Alice A's Password
Otherwise, repeats until hits
Successful Password

Fig 4: Undetectable On-line Password Guessing Attack on 3P-EKE Yoon-Yoo's Protocol

Step 1: Alice sends the credentials {id_a, id_b, id_t, $E_{Pwda}(M_a)$,$h_t(r_a)$, $f_{Kat}(M_a)$} to Bob.
i.e., A→B: {id_a, id_b, id_t, $E_{Pwda}(M_a)$,$h_t(r_a)$, $f_{Kat}(M_a)$}

Step 2: Bob stores the credentials sent by Alice. Now, Bob predicts a password $Pwd_{a'}$ to get $M_{a'}$ by decrypting $E_{Pwda}(M_a)$.

Step 3: Now, Bob selects two random numbers $r_b \in_R Z_p$, to compute $K_{at}=M_{a'}^{rb}$mod p. Bob encrypts $M_{a'}$ with his password Pwd_b and also computes $h_S(r_b)$ & $f_{Kbt}(M_{a'})$. Finally, Bob transfers the credentials {id_a, id_b, id_t, $E_{Pwda}(M_a)$,$h_t(r_a)$, $f_{Kat}(M_a)$,$E_{Pwdb}(M_{a'})$,$h_t(r_b)$, $f_{Kbt}(M_{a'})$ } to Trusted Party.
i.e., B→T : {id_a, id_b, id_t, $E_{Pwda}(M_a)$,$h_t(r_a)$, $f_{Kat}(M_a)$,$E_{Pwdb}(M_{a'})$,$h_t(r_b)$, $f_{Kbt}(M_{a'})$}

Step 4: After getting the credentials sent form Bob, Trusted Party authenticates Alice and Bob by verifying $f_{Kat}(M_a)$ and $f_{Kbt}(M_a^{'})$. If they are valid, Trusted Party by using its trapdoor t get the values r_a & r_b from $h_t(r_a)$ & $h_t(r_b)$ and chooses a random number $RE_t \in_R Z_p$ to compute $(M_a^{'})^{REt}$mod p and $(M_a)^{REt}$mod p and then find the values $M_a^{REt}\oplus r_b$ and $M_{a'}^{REt}\oplus r_{a.}$ Finally, Trusted Party sends the credentials {$M_a^{REt}\oplus$ $r_{b,}$ $f_{Kbt}(id_a, id_{b,} K_{bt}, M_a^{REt})$, $M_{a'}^{REt}\oplus r_a$, $f_{Kat}(id_a, id_{b,} K_{at}, M_{a'}^{REt})$} to Bob.

i.e., T→B: $\{M_a^{REt} \oplus r_b, f_{Kbt}(id_a, id_b, K_{bt}, M_a^{REt}), M_{a'}^{REt} \oplus r_a, f_{Kat}(id_a, id_b, K_{at}, M_{a'}^{REt})\}$

Step 5: First of all, Bob uses r_b to compute $M_a^{REt} \oplus r_b \oplus r_{b=} M_a^{REt}$ and $r_{a'} = M_a^{REt} \oplus M_{a'}^{REt} \oplus r_{a.}$ Then Bob uses $r_{a'}$ to compute $K_{at'}=(M_{a'})^{ra'}(\text{mod } p)$ and $f_{Kat'}(id_a, id_b, K_{at}, M_{a'}^{REt})$. At last, Bob compares the computed results with the received $f_{Kat}(id_a, id_b, K_{at}, M_a^{REt})$. If they are equal, then Bob succeeded in guessing the Alice's password. Otherwise, Bob will repeat the Steps 2 to 5 till match is found. Hence, undetectable on-line dictionary attacks can be easily mounted on Yoon and Yoo's protocol.

V. PROBE OF PSRJ PROTOCOL

In this section, we first review the PSRJ protocol and then how it is cryptanalyzed by Archana et al. i.e., how it is prone to detectable on-line dictionary attack has shown.

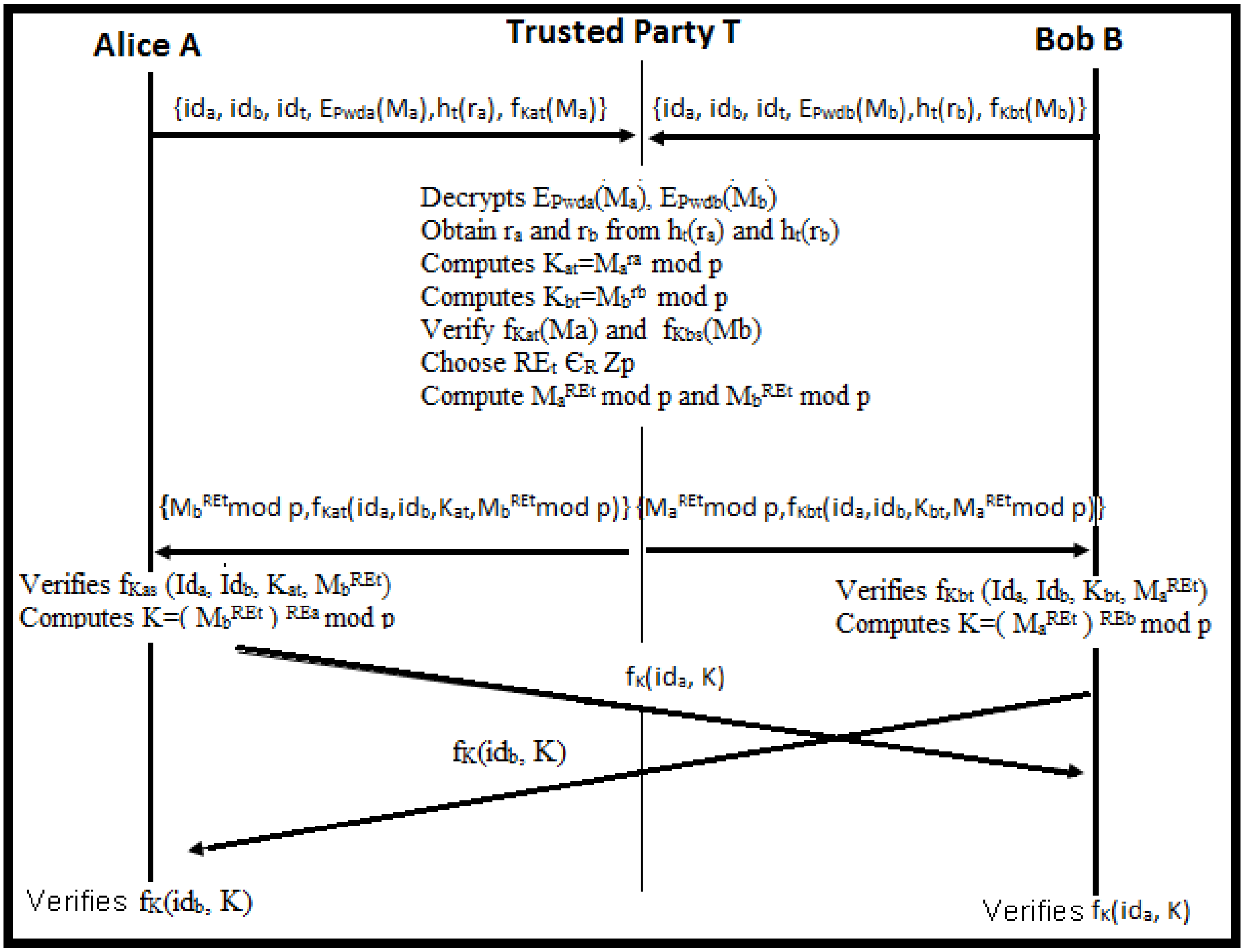

Alice A
Trusted Party T
Bob B
$\{id_a, id_b, id_t, E_{Pwda}(M_a), h_t(r_a), f_{Kat}(M_a)\}$
$\{id_a, id_b, id_t, E_{Pwdb}(M_b), h_t(r_b), f_{Kbt}(M_b)\}$
Decrypts $E_{Pwda}(M_a)$, $E_{Pwdb}(M_b)$
Obtain r_a and r_b from $h_t(r_a)$ and $h_t(r_b)$
Computes $K_{at}=M_a^{ra}$ mod p
Computes $K_{bt}=M_b^{rb}$ mod p
Verify $f_{Kat}(Ma)$ and $f_{Kbs}(Mb)$
Choose $RE_t \in_R Zp$
Compute M_a^{REt} mod p and M_b^{REt} mod p
$\{M_b^{REt}\text{mod } p, f_{Kat}(id_a, id_b, K_{at}, M_b^{REt}\text{mod } p)\}$
$\{M_a^{REt}\text{mod } p, f_{Kbt}(id_a, id_b, K_{bt}, M_a^{REt}\text{mod } p)\}$
Verifies $f_{Kas}(Id_a, Id_b, K_{at}, M_b^{REt})$
Computes $K=(M_b^{REt})^{REa}$ mod p
Verifies $f_{Kbt}(Id_a, Id_b, K_{bt}, M_a^{REt})$
Computes $K=(M_a^{REt})^{REb}$ mod p
$f_K(id_a, K)$
$f_K(id_b, K)$
Verifies $f_K(id_b, K)$
Verifies $f_K(id_a, K)$

Fig 5: PSRJ 3P-EKE Protocol

A. *Review*

The details of the PSRJ protocol are illustrated in Fig 5. For instance, the initial credentials sent from Alice and Bob are same as in Yoon-Yoo's protocol. The steps of the protocol are as follows:

Step 1: Alice sends the credentials $\{id_a, id_b, id_t, E_{Pwda}(M_a), h_t(r_a), f_{Kat}(M_a)\}$ to Trusted Party. Simultaneously, Bob sends the credentials $\{id_a, id_b, id_t, E_{Pwdb}(M_b), h_t(r_b), f_{Kbt}(M_b)\}$ to Trusted Party.
i.e., A→T: $\{id_a, id_b, id_t, E_{Pwda}(M_a), h_t(r_a), f_{Kat}(M_a)\}$ and
B→T: $\{id_a, id_b, id_t, E_{Pwdb}(M_b), h_t(r_b), f_{Kbt}(M_b)\}$
Step 2: After receiving the credentials from Alice & Bob, a Trusted Party verifies it by computing the values same as in the Chang-Chang's protocol. Then a Trusted Party computes M_b^{REt} mod p & M_a^{REt} mod p and sends the credentials $\{M_b^{REt}\text{mod } p, f_{Kat}(id_a, id_b, K_{at}, M_b^{REt}\text{mod } p)\}$ to Alice and $\{M_a^{REt}\text{mod } p, f_{Kbt}(id_a, id_b, K_{bt}, M_a^{REt}\text{mod } p\}$ to Bob.
i.e., T→A: $\{M_b^{REt}\text{mod } p, f_{Kat}(id_a, id_b, K_{at}, M_b^{REt}\text{mod } p)\}$ and
T→B: $\{M_a^{REt}\text{mod } p, f_{Kbt}(id_a, id_b, K_{bt}, M_a^{REt}\text{mod } p)\}$

Step 3: Bob calculates the session key $K= (M_a^{REt})^{REb}$ mod p and sends $f_K(id_b, K)$ to Alice.

i.e., B →A: $f_K(id_b, K)$
Step 4: Alice calculates the session key K= ($M_b^{REt})^{REa}$ mod p and sends $f_K(id_a, K)$ to Bob.
i.e., A →B: $f_K(id_a, K)$

After verifying the received messages $f_K(id_b, K)$ & $f_K(id_a, K)$, Alice & Bob can confirms that they actually share a session key K= $(M_b^{REt})^{REa}$ (mod p) = $(M_a^{REt})^{REb}$ (mod p) at present. Otherwise, the current session of the protocol will be terminated.

B. Attack

This section exhibits a detectable on-line dictionary attack on PSRJ protocol by Archana et al. The details of an attack are illustrated in Fig 6. The details of an attack are shown below:

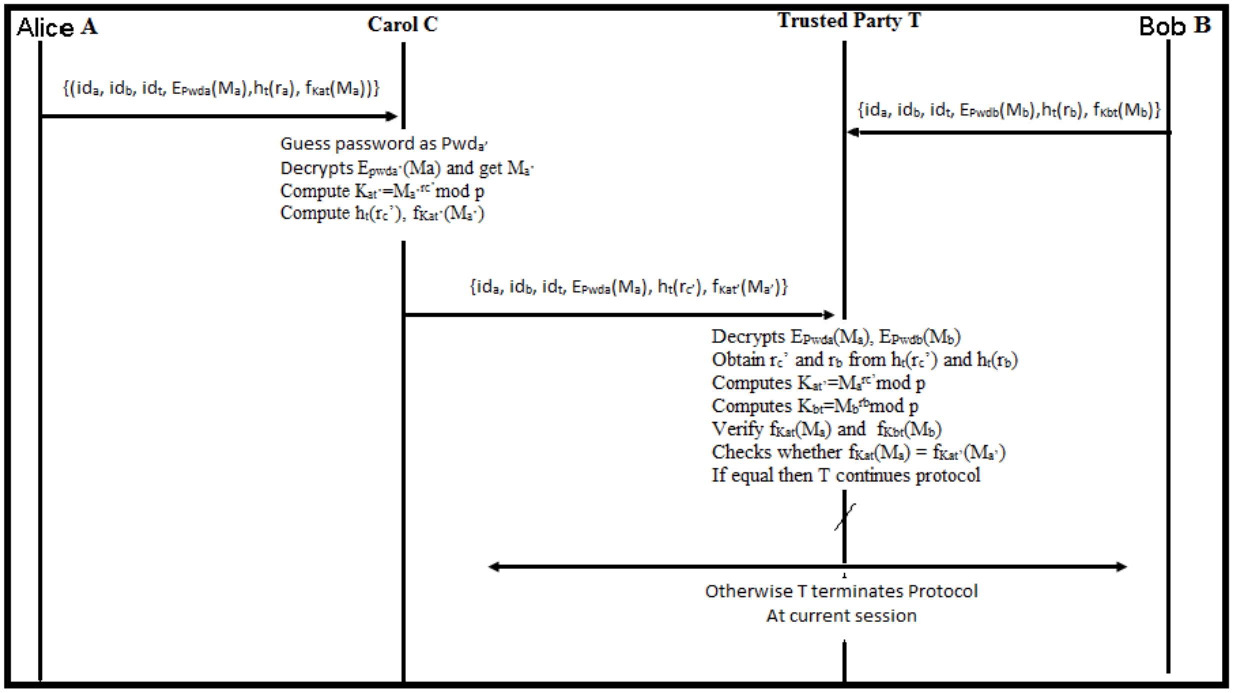

Alice A
Carol C
Trusted Party T
Bob B
{(id_a, id_b, id_t, $E_{Pwda}(M_a)$,$h_t(r_a)$, $f_{Kat}(M_a)$)}
{id_a, id_b, id_t, $E_{Pwdb}(M_b)$,$h_t(r_b)$, $f_{Kbt}(M_b)$}
Guess password as $Pwd_{a'}$
Decrypts $E_{pwda'}$(Ma) and get $M_{a'}$
Compute $K_{at'}=M_{a'}^{rc'}$mod p
Compute $h_t(r_c')$, $f_{Kat'}(M_{a'})$
{id_a, id_b, id_t, $E_{Pwda}(M_a)$, $h_t(r_{c'})$, $f_{Kat'}(M_{a'})$}
Decrypts $E_{Pwda}(M_a)$, $E_{Pwdb}(M_b)$
Obtain r_c' and r_b from $h_t(r_c')$ and $h_t(r_b)$
Computes $K_{at'}=M_a^{rc'}$mod p
Computes $K_{bt}=M_b^{rb}$mod p
Verify $f_{Kat}(M_a)$ and $f_{Kbt}(M_b)$
Checks whether $f_{Kat}(M_a) = f_{Kat'}(M_{a'})$
If equal then T continues protocol
Otherwise T terminates Protocol
At current session

Fig 6: Detectable Online Dictionary Attack on PSRJ 3P-EKE Prtocol

An invader Carol can mimic Alice and communicate with Bob. While Bob is thinking that it is interacting with Alice but actually it is talking with an invader Carol.

Step 1: Alice selects two random numbers viz., RE_a ,$r_a \in_R Z_p$ and calculates $E_{Pwda}(M_a)$, $h_t(r_a)$ and $f_{Kat}(M_a)$,where M_a=g ^{REa}mod p and $K_{at}=M_a^{ra}$mod p .Then she sends {(id_a, id_b, id_t, $E_{Pwda}(M_a)$,$h_t(r_a)$, $f_{Kat}(M_a)$)} to Trusted Party.

i.e., A→T : {(id_a, id_b, id_t, $E_{Pwda}(M_a)$,$h_t(r_a)$, $f_{Kat}(M_a)$)}

Step 2: An invader Carol intercepts this message i.e., {(id_a, id_b, id_t, $E_{Pwda}(M_a)$,$h_t(r_a)$, $f_{Kat}(M_a)$)} and guess Alice's password $Pwd_{a'}$ to decrypt $E_{Pwda}(M_a)$ and gets $M_{a'}$. Now she generates a random number $r_{c'} \in_R Z_p$ and computes $K_{at'}=M_{a'}^{rc'}$ mod p. Then she sends {id_a, id_b, id_t, $E_{Pwda}(M_a)$, $h_t(r_{c'})$, $f_{Kat'}(M_{a'})$} to Trusted Party.
i.e., C→T : {id_a, id_b, id_t, $E_{Pwda}(M_a)$, $h_t(r_{c'})$, $f_{Kat'}(M_{a'})$}

Step 3: Upon receiving the credentials {id_a,id_b,id_t,$E_{Pwda}(M_a)$,$h_t(r_{c'})$,$f_{Kat'}(M_{a'})$} from Carol(instead of Alice), Trusted Party decrypts $E_{Pwda}(M_a)$ to get M_a. Then it retrieves $r_{c'}$ from $h_t(r_{c'})$ by using trapdoor[15] t. Now Trusted Party computes $K_{at'}=M_{a'}^{rc'}$ mod p to authenticate the received $f_{Kat'}(M_{a'})$. If both $f_{Kat}(M_a)$ and $f_{Kat'}(M_{a'})$ are equal then the guessed password is correct. So Trusted Party will continue with the remaining residual procedure of the protocol.
For instance, the Trusted Party can detect the attack and it terminates the protocol at current session. An invader never sits indolent. She will continue the same process after some time. She will repeat this process until she hits the successful password. In this way a malevolent user can get a session key successfully by impersonating the actual user.

VI. PROBE OF RAJ ET AL. PROTOCOL

In this section, we first review the Raj et al. protocol and then how it is cryptanalyzed by Archana et al. i.e., how it is vulnerable to detectable on-line dictionary attack has shown.

A. Review:

The protocol is demonstrated in Fig7. The complete details are as follows.

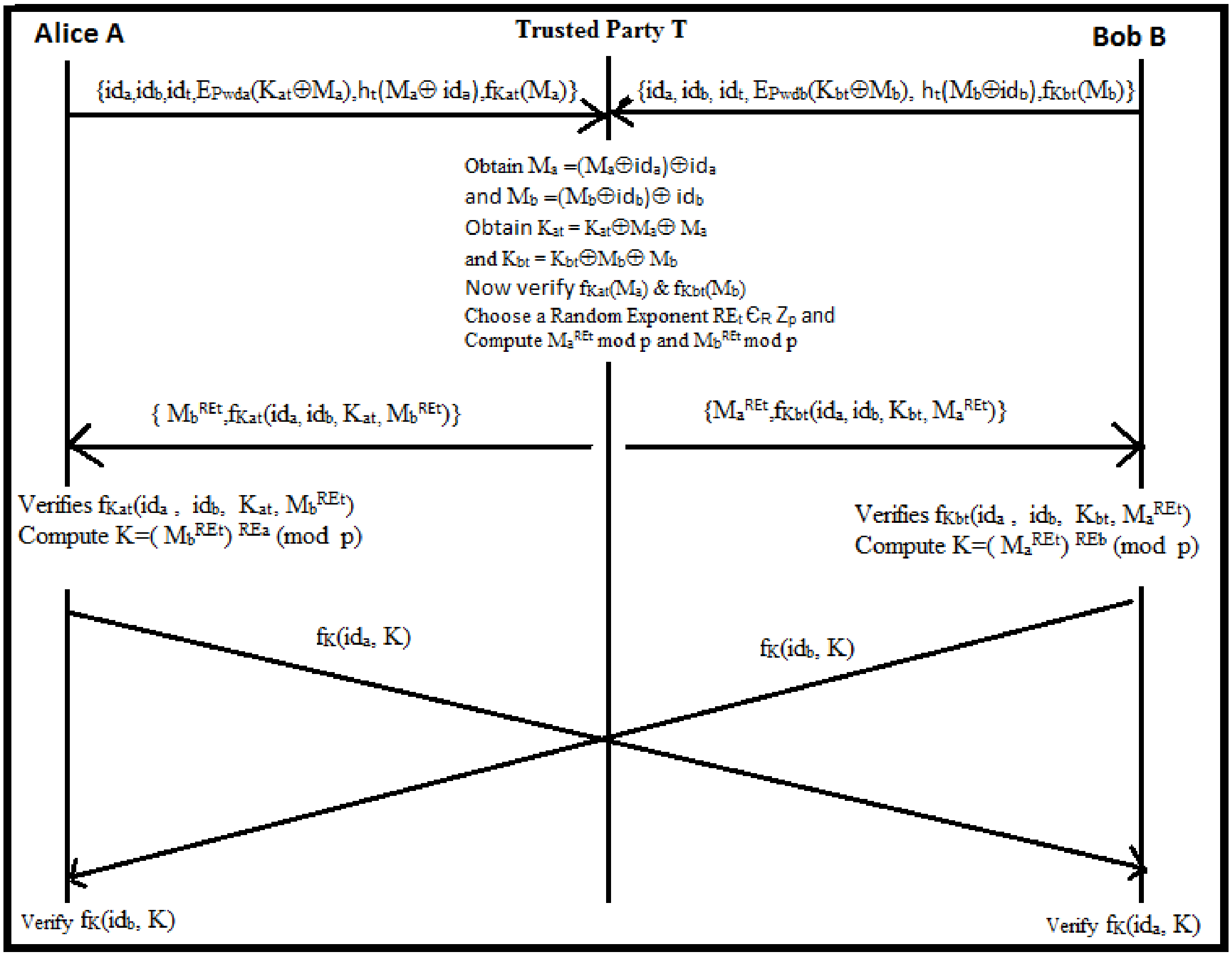

Alice A
Trusted Party T
Bob B
$\{id_a, id_b, id_t, E_{Pwda}(K_{at}\oplus M_a), h_t(M_a\oplus id_a), f_{Kat}(M_a)\}$
$\{id_a, id_b, id_t, E_{Pwdb}(K_{bt}\oplus M_b), h_t(M_b\oplus id_b), f_{Kbt}(M_b)\}$
Obtain $M_a = (M_a\oplus id_a)\oplus id_a$
and $M_b = (M_b\oplus id_b)\oplus id_b$
Obtain $K_{at} = K_{at}\oplus M_a\oplus M_a$
and $K_{bt} = K_{bt}\oplus M_b\oplus M_b$
Now verify $f_{Kat}(M_a)$ & $f_{Kbt}(M_b)$
Choose a Random Exponent $RE_t \in_R Z_p$ and
Compute M_a^{REt} mod p and M_b^{REt} mod p
$\{M_b^{REt}, f_{Kat}(id_a, id_b, K_{at}, M_b^{REt})\}$
$\{M_a^{REt}, f_{Kbt}(id_a, id_b, K_{bt}, M_a^{REt})\}$
Verifies $f_{Kat}(id_a, id_b, K_{at}, M_b^{REt})$
Compute $K = (M_b^{REt})^{REa}$ (mod p)
Verifies $f_{Kbt}(id_a, id_b, K_{bt}, M_a^{REt})$
Compute $K = (M_a^{REt})^{REb}$ (mod p)
$f_K(id_a, K)$
$f_K(id_b, K)$
Verify $f_K(id_b, K)$
Verify $f_K(id_a, K)$

Fig 7: Raj et al. 3P-EKE Protocol

Step 1: Alice selects two random numbers viz., $RE_a, r_a \in_R Z_p$ and calculates $E_{Pwda}(K_{at}\oplus M_a)$, $h_t(M_a\oplus id_a)$ & $f_{Kat}(M_a)$. Now she sends the credentials $\{id_a, id_b, id_t, E_{Pwda}(K_{at}\oplus M_a), h_t(M_a\oplus id_a), f_{Kat}(M_a)\}$ to Trusted Party.

i.e., A→T : $\{id_a, id_b, id_t, E_{Pwda}(K_{at}\oplus M_a), h_t(M_a\oplus id_a), f_{Kat}(M_a)\}$

Simultaneously, Bob also computes $E_{Pwdb}(K_{bt}\oplus M_b)$, $h_t(M_b\oplus id_b)$ and $f_{Kbt}(M_b)$ by generating his own random numbers viz., $RE_b, r_b \in_R Z_p$. and transmits $\{id_a, id_b, id_t, E_{Pwdb}(K_{bt}\oplus M_b), h_t(M_b\oplus id_b), f_{Kbt}(M_b)\}$ to Trusted Party.

i.e., B→T : $\{id_a, id_b, id_t, E_{Pwdb}(K_{bt}\oplus M_b), h_t(M_b\oplus id_b), f_{Kbt}(M_b)\}$

Here, Alice and Bob simultaneously communicate with the Trusted Party.

Step 2: After receiving the messages sent from Alice & Bob , a Trusted Party uses a trapdoor t to get $M_a\oplus id_a$ and $M_b\oplus id_b$ from $h_t(M_a\oplus id_a)$ and $h_t(M_b\oplus id_b)$ and then retrieves $M_a=(M_a\oplus id_a)\oplus id_a$ and $M_b=(M_b\oplus id_b)\oplus id_b$ respectively. Now, a Trusted Party by using the passwords of Alice & Bob respectively, decrypts $E_{Pwda}(K_{at}\oplus M_a)$ & $E_{Pwdb}(K_{bt}\oplus M_b)$ and gets $K_{at}\oplus M_a$ & $K_{bt}\oplus M_b$ to compute $K_{at} = K_{at}\oplus M_a\oplus M_a$ and $K_{bt} = K_{bt}\oplus M_b\oplus M_b$. Now, Trusted Party computes $f_{Kat}(M_a)$ and $f_{Kbt}(M_b)$ and verifies whether computed value $f_{Kat}(M_a)$ (or $f_{Kbt}(M_b)$) and received value $f_{Kat}(M_a)$ (or $f_{Kbt}(M_b)$)) are identical or not.

If identical, then a Trusted Party continues with the remaining steps of the protocol. Subsequently, a Trusted Party computes M_a^{REt}mod p & M_b^{REt}mod p and then corresponding hashed credentials $f_{Kat}(id_a, id_b, K_{at}, M_a^{REt})$ & $f_{Kbt}(id_a, id_b, K_{bt}, M_a^{REt})$.Finally, a Trusted Party sends { M_b^{REt}, $f_{Kat}(id_a, id_b, K_{at}, M_b^{REt})$} to Alice and { M_a^{REt},$f_{Kbt}(id_a, id_b, K_{bt}, M_a^{REt})$} to Bob simultaneously.

i.e., **T→A:** $\{M_b^{REt}, f_{Kat}(id_a, id_b, K_{at}, M_b^{REt})\}$

T→B: $\{M_a^{REt}, f_{Kbt}(id_a, id_b, K_{bt}, M_a^{REt})\}$

If not equal, then a Trusted Party ends the protocol at present session.

Step 3: Upon receiving the messages sent from a Trusted Party, Bob first verifies $f_{Kbt}(id_a, id_b, K_{bt}, M_a^{REt})$ to

authenticate a Trusted Party. If this authentication is approved, Bob trusts the received M_a^{REt} is legal. Then he computes a session key K=($M_a^{REt})^{REb}$ (mod p) & $f_K(id_b, K)$ and sends it to Alice.
i.e., B →A: $f_K(id_b, K)$.
If authentication is failed, then a user Bob ends the protocol.

Step 4: Upon receiving the messages sent from a Trusted Party, Alice first verifies $f_{Kat}(id_a, id_b, K_{at}, M_b^{REt})$ to authenticate a Trusted Party. If this validation holds Alice computes the session key K= ($M_b^{REt})^{REa}$ (mod p). Thereafter, Alice verifies Bob by checking the computed $f_K(id_b, K)$ with the received one. If equal, then she computes $f_K(id_a, K)$ and sends it to Bob.

A → B: $f_K(id_a, K)$

If any one of the authentication process fails then a user Alice terminates the protocol.

Step 5: After successfully examining the received messages $f_K(id_b, K)$ & $f_K(id_a, K)$ repectively, both Alice & Bob confirms that they truly share a secret session key K= ($M_b^{REt})^{REa}$mod p= $(M_a^{REt})^{REb}$ mod p at present session. Otherwise, the protocol will be terminated.

B. Attack

In this section, it is shown that how Archana et al. cryptanalyzed the Raj et al. 3P-EKE protocol. Carol an intruder mimics Alice and communicates with Bob. While Bob is thinking that it is interacting with Alice but actually it is talking with Carol.

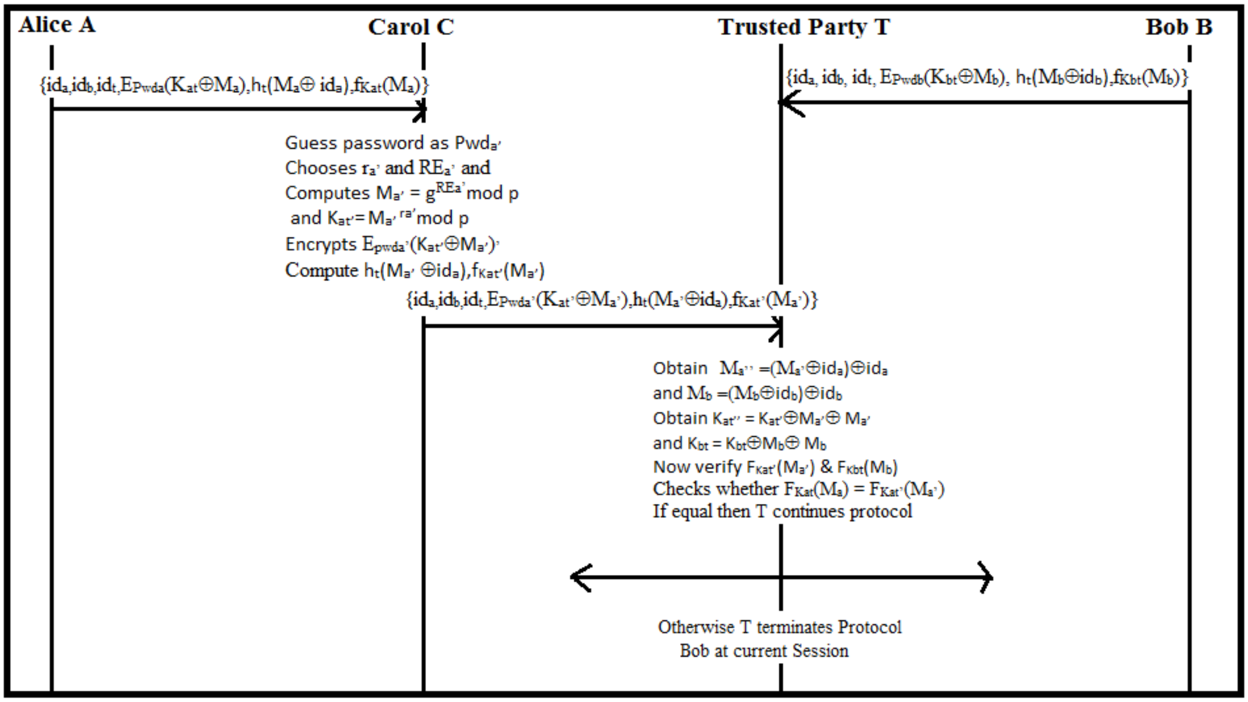

Alice A
Carol C
Trusted Party T
Bob B
{ida,idb,idt,EPwda(Kat⊕Ma),ht(Ma⊕ ida),fKat(Ma)}
{ida, idb, idt, EPwdb(Kbt⊕Mb), ht(Mb⊕idb),fKbt(Mb)}
Guess password as Pwda'
Chooses ra' and REa' and
Computes Ma' = gREa' mod p
and Kat' = Ma' ra' mod p
Encrypts Epwda'(Kat'⊕Ma')
Compute ht(Ma' ⊕ida),fKat'(Ma')
{ida,idb,idt,EPwda'(Kat'⊕Ma'),ht(Ma'⊕ida),fKat'(Ma')}
Obtain Ma'' =(Ma'⊕ida)⊕ida
and Mb =(Mb⊕idb)⊕idb
Obtain Kat'' = Kat'⊕Ma'⊕ Ma'
and Kbt = Kbt⊕Mb⊕ Mb
Now verify FKat'(Ma') & FKbt(Mb)
Checks whether FKat(Ma) = FKat'(Ma')
If equal then T continues protocol
Otherwise T terminates Protocol
Bob at current Session

Fig 8: Detectable Online Ditionary Attack on Raj et al 3P-EKE Protocl

Detectable on-line password guessing attack on Raj et al. 3P-EKE Protocol is illustrated in Fig 8. The details of the attack are shown below.

Step 1: Alice generates two random numbers viz., RE_a , r_a $\in_R Z_p$ and calculates $E_{Pwda}(K_{at}\oplus M_a)$, $h_t(M_a\oplus id_a)$ & $f_{Kat}(M_a)$, where $M_a=g^{REa}$(mod p) and $K_{at}=M_a^{ra}$ (mod p). Then she sends {id_a, id_b, id_t, $E_{Pwda}(K_{at}\oplus M_a)$, $h_t(M_a\oplus id_a)$, $f_{Kat}(M_a)$} to Trusted Party.
i.e., A→T: {id_a, id_b, id_t, $E_{Pwda}(K_{at}\oplus M_a)$, $h_t(M_a\oplus id_a)$, $f_{Kat}(M_a)$}

Step 2: An invader Carol intercepts the message {id_a, id_b, id_t, $E_{Pwda}(K_{at}\oplus M_a)$, $h_t(M_a\oplus id_a)$, $f_{Kat}(M_a)$} and generates two random numbers viz., $R_{Ea'}$,$r_{a'}$$\in_R Z_p$ and computes $M_{a'}=g^{REa'}$(mod p) & $K_{at'}= M_{a'}^{ra'}$(mod p). Now Invader Carol guess Alice's password as $Pwd_{a'}$ to encrypt $(K_{at'}\oplus M_{a'})$. Next, she computes the another two credentials $h_t(M_{a'}\oplus id_a)$, $f_{Kat'}(M_{a'})$ as the id's are not secret. Then she sends the credentials {id_a, id_b, id_t, $E_{Pwda'}(K_{at'}\oplus M_{a'})$, $h_t(M_{a'} \oplus id_a)$, $f_{Kat'}(M_{a'})$} to a Trusted Party.
i.e., C→T: {id_a, id_b, id_t, $E_{Pwda'}(K_{at'}\oplus M_{a'})$, $h_t(M_{a'} \oplus id_a)$, $f_{Kat'}(M_{a'})$}
Step 3: After receiving the credentials {id_a, id_b, id_t, $E_{Pwda'}(K_{at'}\oplus M_{a'})$, $h_t(M_{a'}\oplus id_a)$, $f_{Kat'}(M_{a'})$}, a Trusted Party decrypts $E_{Pwda'}(K_{at'}\oplus M_{a'})$ to get $(K_{at'}\oplus M_{a'})$. Then it retrieves $(M_{a'}\oplus id_a)$ from $h_t(M_{a'}\oplus id_a)$ by using trapdoor t. Now, Trusted Party computes $M_{a''}=(M_{a'}\oplus id_a)\oplus id_a$ to obtain $K_{at''}= (K_{at'}\oplus M_{a'})\oplus M_{a''}$. Next, Trusted Party verifies whether computed $f_{Kat''}(M_{a''})$ and received $f_{Kat'}(M_{a'})$ are equal or not.

If both $f_{Kat''}(M_{a''})$ and $f_{Kat'}(M_{a'})$ are equal then the predicted password is correct and Trusted Party will continue the residual procedure of the protocol.

If not equal, then the attack is detected by Trusted Party and terminates the protocol at a current session. An intruder never sits idle. After some time she repeats the same process. She will continue with this process until she hits the successful password. In this way a malicious user can impersonate the actual user by successfully getting a secrete session key.

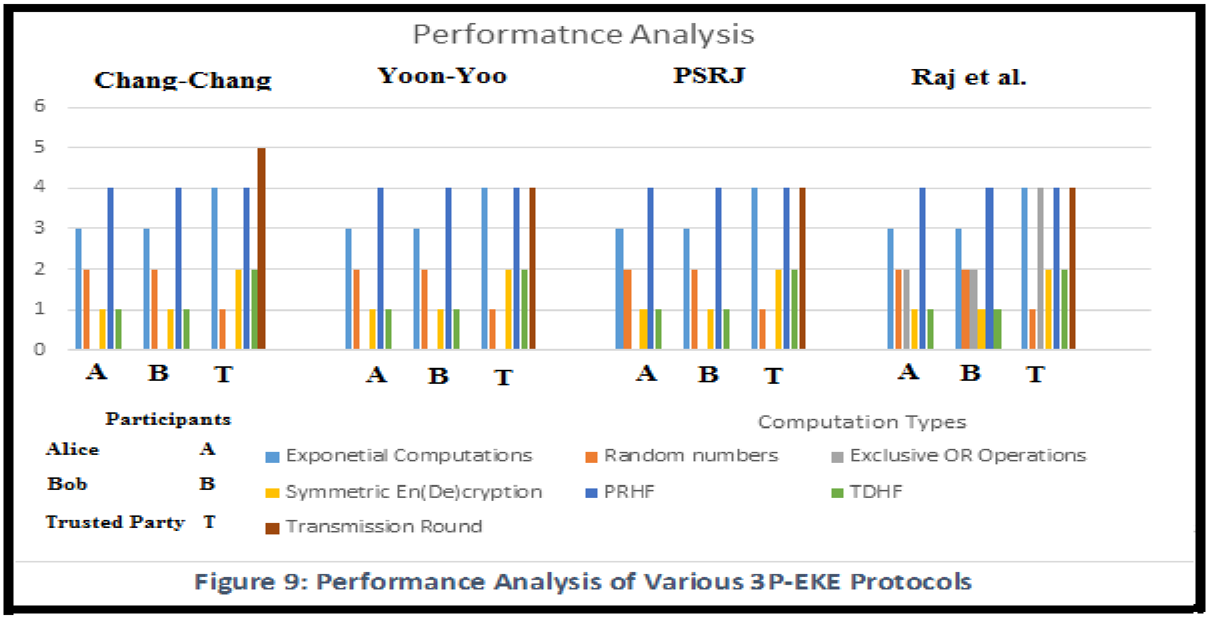


Figure 9: Performance Analysis of Various 3P-EKE Protocols

3P-EKE Protocols → Participants ▲ Computation Type & Attack Type	Chang-Chang's Protocol			Yoon-Yoo's, Protocol			PSRJ protocol			Raj et.al protocol		
	Alice	Bob	Trusted Party	Alice	Bob	Trusted Party	Alice	Bob	Trusted Party	Alice	Bob	Trusted Party
Exponential Computations	3	3	4	3	3	4	3	3	4	3	3	4
Random numbers	2	2	1	2	2	1	2	2	1	2	2	1
Exclusive OR Operations	0	0	0	0	0	0	0	0	0	2	2	4
Symmetric en(de)cryption	1	1	2	1	1	2	1	1	2	1	1	2
Pseudo Random hash Function(PRHF)Operations	4	4	4	4	4	4	4	4	4	4	4	4
Trapdoor hash functions (TDHF) Operations	1	1	2	1	1	2	1	1	2	1	1	2
Transmission Round	5			4			4			4		
Undetectable On-line Password Guessing Attack	YES			YES			YES			YES		
Detectable On-line password Guessing Attack	YES			YES			YES			YES		
Off-line Password Guessing Attack	YES			YES			YES			YES		

Table 2. Performance Analysis of various 3P-EKE Protocols

VII. PERFORMANCE ANALYSIS

In order to determine the efficiency of a protocol, the number of transmission rounds and computational complexity can be taken into account. The comparison of computational complexities among four 3P-EKE Protocols viz., Chang-Chang's , Yoon-Yoo's , PSRJ and Raj et.al protocols are depicted in Fig 9.

The comparison of different types of attacks among the four 3P-EKE Protocols viz.,Chang-Chang, Yoon-Yoo's , PSRJ and Raj et.al protocols are shown in Table 2. From this table we can conclude that all the above protocols are vulnerable to all types of attacks explained by Ding & Horster.

VIII. CONCLUSION

To establish a secure communication within an untrusted network, password-based authenticated encrypted key exchange protocols (PAEKE) is widely setup due its simplicity and convenience of maintaining a low entropy password at user side on lots of remote user authentication system. In password authenticated 3P-EKE (third-party encrypted key exchange) protocols, users share a human memorable passwords with the trusted third party to establish a secure secret session key for further communication via insecure channel. Such 3P-EKE protocols can be used for applications in which light-weight users wants to communicate securely. In this paper we have investigated four 3P-EKE Protocols. Initially, in Section III we reviewed Chang-Chang's ECC-3PEKE which is based on without using the server's public keys. They claimed that their proposed ECC-3PEKE protocol is secure, efficient, and practical. Unlike their claims, the ECC-3PEKE protocol however, is still vulnerable to undetectable on-line password guessing attacks. Accordingly, the current paper demonstrates the Chang-Chang's protocol and its vulnerability exposed by Yoon and Yoo. In Section IV we probed an improved protocol proposed by AYoon and Yoo based on Exclusive-OR operations. But unfortunately, Padmavathy et al. has shown that still it suffers from undetectable online dictionary attack. In next section, we have investigated an enhanced protocol (PSRJ) which is proposed by Padmavathy et al. They have proved that this protocol (PSRJ) could achieve better performance efficiency because it requires only two message transmission rounds. Subsequently, it is cryptanalyzed by Archana et al. They proved that how this protocol is exposed to detectable on-line password guessing attack by successfully getting the secret session key. Later in section VI, we first reviewed the Raj et al. protocol in which the parallel transmission technique is used to improve the efficiency along with XOR operations. Unfortunately, it is cryptanalyzed by Archana et al. to get the secret session key successfully. Finally, we compared the computation complexities in terms of PRHF, TDHF, Exponential computation, XOR operations etc. and different types of attacks may suffer from for all of these four 3P-EKE protocols.

REFERENCES

[1] Chin-Chen Chang*, Ya-fen Chang, "A novel three-party encrypted key exchange protocol", *Elsevier, Computer Standards & Interfaces,* Volume-26, Issue/No-5, Page No (471 – 476), 2004.

[2] Y. Ding and P. Horster, "Undetectable on-line password guessing attacks," *ACM Operating Systems Review*, Volume-9, Issue/No-4, Page no (77-86), 1995.

[3] W. Diffie and M. E. Hellman. "New directions in cryptography". IEEE Transactions on Information Theory, Volume-22, Issue/No-6, Page No (644– 654), 1976.

[4] S.M. Bellovin and M. Merritt, "Encrypted key exchange: password-based protocols secure against password guessing attacks," *in Proc. of 1992 IEEE Symposium on Research in Security and Privacy*, Page No (72–84), 1992.

[5] M. Steiner, G. Tsudik, and M. Waidner, "Refinement and extension of encrypted key exchange", *ACM Operating Systems Review*, Volume-29, Issue/No-5, Page No (22-30), 1995.

[6] C. L. Lin, H. M. Sun, and T. Hwang, "Three party encrypted key exchange: attacks and a solution", *ACM Operating Systems Review*, Volume -34, Issue/No-4, Page No (12-20), 2002.

[7] Feng Zhu, Duncan S. Wong, Agnes H. Chan, Robbie Ye "Password Authenticated Key Exchange Based on RSA for Imbalanced Wireless Networks" *Information Security, Lecture Motes in Computer Science,* Volume -2433, Page No (150-161), 2002.

[8] Yeh H-T, Sun H-M, Hwang T, "Efficient three-party authentication and key agreement protocols resistant to password guessing attacks", ***Journal of Information Science and Engineering,*** Volume-19, Issue/No-6, Page No (1059-1070), 2003.

[9] T. f. Lee, T. Hwang, and C. L. Lin, "Enhanced three party encrypted key exchange without server public Keys". *Computers and Security*, Volume-23, Issue/No-7, Page No (571-577), 2004.

[10] Eun-Jun Yoon, Kee-Young Yoo, "Improving the novel three-party encrypted key exchange protocol", Elsevier, *Computer Standards and Interfaces*, Volume-30, Issue/No-5, Page No (309-314) , 2008.

[11] R.Padmavathy , Tallapally Shirisha , M.Rajkumar , JayadevGyani, "Improved analysis on Chang and Chang Password Key Exchange Protocol", *IEEE International Conference on Advances in Computing, Control, and Telecommunication Technologies*, Page No (781-783), 2009.

[12] Archana Raghuvamshi, P.Venkateshwara Rao,and Prof.P.Premchand , "Cryptanalysis of Authenticated Key Exchange 3P-EKE Protocol and its Enhancement", *IEEE-International Conference On Advances In Engineering, Science And Management (ICAESM -2012)*, Page No (659-666), March 30-31, 2012.

[13] P.Rajkumar, C.Manoharan, M.Ananthi, "Performance Analysis Of 3pek Exchange Protocol Using Parallel Message Transmission Technique", *International Journal of Engineering Research & Technology (IJERT)*, Volume-1, Issue/No-7, Page No (1-5) Sep 2012.

[14] Archana Raghuvamshi , Prof.P.Premchand ,"A Weakness in 3pek Exchange Protocol using Parallel Message Transmission Technique", *International Journal of Advanced Research in Computer Science(IJARCS)*, Volume-4, Issue/No-11, Page No (104-108), Dec 2013

[15] Y. Gertner, T. Malkin, O. Reingold, "On the impossibility of basing trapdoor functions on trapdoor predicates", *Proceedings of the 42nd IEEE Symposium on foundations of Computer Science*, Las Vegas, Mevada, Page No (126 – 135), Oct 2001.

AUTHORS PROFILE

Authors' Profiles

Corresponding Author:

Name: ARCHANA RAGHUVAMSHI

Academic Achievements:

She received her Bachelor's Degree BSc (M.S.Cs), Master's Degrees M.C.A and M.Tech(CSE) from Osmania University, Hyderabad. She did course work in ADS and WMN in IITM (Indian Institute of Technology, Madras). She is perusing PhD (CSE) in JNTUK, Kakinada. She has published four research papers in IEEE Digital library and another four research papers in various peer reviewed International Journals. Her research interest includes Cryptography and Information Security, Security in Cloud Computing etc.

Professional Bodies:

She is a,

1. Professional Member of ACM
2. Member of Professional Body IAENG
3. Member of IACSIT
4. Associate Member of theIRED

Work Experience:

She is having 13+ year of teaching experience. She is working as an Assistant Professor in Dept. of CSE, UCOE, Adikavi Nannaya University, Rajahmundry, India.

Co-Author:

Name: PROF.PREMCHAND PARVATANENI

Academic Achievements:

He received his Bachelor's Degree B.Sc (Engg.) from RIT, Jamshedpur. He received his Master's M.E (CE) from AU (Andhra University), Visakhapatnam. He received his PhD(CSSE) from AU. He has published more than 50 publications in various International Journals and Conference proceedings. His research Interest includes Cryptography and Network Security, Image Processing, Software Engineering etc.

Work Experience:

He is having 40+ years of teaching experience in various Universities. He is working as a Professor in Dept. of CSE, University College of Engineering, Osmania University. He was as a Director in AICTE, New Delhi. And also he has been held with the various positions like Head, Chairman of BOS, Additional Controller of Examinations in Professional wing, Osmania University, Hyderabad.