A Password Attack on S-3 PAKE

R. Padmavathy

Dept of Computer Science and Engineering
National Institute of Technology, Warangal, India
r_padma3@rediffmail.com

Abstract

The key exchange protocols using passwords achieved great attention due to their simplicity and e ciency. On the other hand these protocols should resist all types of password guessing attacks, as the password is of low entropy. Recently Lu and Cao proposed a three party simple key ex-change protocol (S-3PAKE). It is an extension of Abdella and Pointchval SPAKE protocol. Later Guo et al. proposed a man in the middle attack and undetectable on-line password guessing attack on the above protocol. They also presented an improved S-3PAKE. In the present paper we have shown that the improved S-3PAKE still su ers from undetectable pass-word guessing attack and discuss the countermeasures to preclude such an attack.

Keywords: *PAKE,S-3PAKE,improved S-3PAKE*

1. Introduction

The key exchange protocol is one of the most elegant ways of establishing se-cure communication between a 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 was proposed by Di e-Hellman[4]. Since the introduction of key exchange protocol by Di e-Hellman, various versions and improvements in key exchange protocol have been developed. In the line of key exchange protocol developments, pass-word based key exchange mechanism achieved attention due to its simplicity and wide range of applicability, as it requires the users to remember the eas-ily rememberable password [3, 7, 8]. Even though the protocol is simple and e cient, according to Ding and Hoster[5], it should not be vulnerable to any type of o -line, undetectable or detectable on-line password guessing attacks, as the passwords are of low-entropy.

In general the password guessing attacks can be divided into the three classes 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 cor-rectness 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 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.
- On-line password guessing attacks : An attacker guesses a password and verifies

his/her guess on-line. No participation of server is required, so the server does not notice the attack.

Since the first proposal of Bellovin and Merrit (PAKE) [2], many e - cient key exchange protocols based on password have been developed. Recently these two party key exchange protocols are extended to three party, in which, the two parties initially communicate the passwords with the trusted server se-curely. 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[10]. Subsequently Ding and Hoster published on-line and o -line guessing attacks on Stener's protocol [5]. Later lin et al. proposed two versions of im-proved three party protocol[7], one with server's public key and another without. Recently Chang and Chang [3] proposed a novel three party encrypted key exchange protocol without server public key and claimed the pro-tocol is secure, e cient and practical. Unlike their claims Yoon and Yoo [11] pointed out an undetectable password guessing attack on their protocol, in which one party is able to know the other party's password and furthermore they presented an improved version of it to avoid the above attack. In the sim-ilar line Lu and Cao [8] extended the Adbella and Pointcheval protocol (SPAKE)[1] and proposed a simple three party key exchange protocol (S-3PAKE). More recently Phan et al. [9] pointed out the unknown key share attack and undetectable password guessing attack on S-3PAKE and Guo et al. [6] proposed the man in the middle attack and undetectable on line dictionary attack. They also presented an improved version of S-3PAKE protocol to resist the above attack.

In this paper we have shown that the improved S-3PAKE [6] still suffers from the undetectable on-line dictionary attack. The countermeasures which they suggested do not fully solve the on-line dictionary attack. In the present work we gave another remedy to preclude the above attack.

The paper is organized as follows: section 2 briefly presents the S-3PAKE protocol, section 3 presents the undetectable password guessing attack on S-3PAKE and improved S-3PAKE, section 4 discusses the online undetectable password guessing attack and the countermeasures to avoid the attack and finally the concluding remarks are made in section 5.

2. Review of S-3PAKE

In this section we review the simple three party key exchange protocol (S-3PAKE).

2.1 Simple three party key exchange protocol

The notations used in the protocol are given below:

(G, g, p) : a finite cyclic group G generated by an element g of prime order p.
M, N: two elements in G
S: a trusted server
A, B: two clients
pw1: the password shared between A and S
pw2: the password shared between B and S
H, H′: two secure one-way hash functions.

Procedure to be followed in the three party key exchange protocol

Step 1:

A1: A chooses a random number $x \in Z_p$ and computes $X \leftarrow g^x N^{pw1}$, and then sends A||X to B.

B1: B also chooses a random number $y \in Z_p$ and computes $Y \leftarrow g^y N^{pw2}$, then sends A||X ||B||Y to S.

Step 2:

S2: Upon receiving A||X ||B||Y , the server S first uses the passwords pw1 and pw2 to compute $g^x \leftarrow X / M^{pw1}$ and $g^y \leftarrow Y / N^{pw2}$, respectively. Then she chooses another random number $z \in Z_p$ and computes $g^{zx} \leftarrow (g^z)^x$, $g^{yx} \leftarrow (g^z)^y$. Finally, she sends X′||Y′ to B, where $X' \leftarrow g^{yz} H(A, S, g^x)^{pw1}$ and $Y' \leftarrow g^{xz} H(B, S, g^y)^{pw2}$.

B2: when B receives X′||Y′, he uses the password pw2 to compute $g^{xz} \leftarrow Y' / g^{xz}.H(B, S, g^y)^{pw2}$, and uses a random number y to compute $g^{xyz} \leftarrow (g^{xz})^y$.

At last, he forwards X′||α to A, where $\alpha \leftarrow H(A, B, g^{xyz})$.

Step 3:

A3: After A receives X′||α, she first computes $g^{yz} \leftarrow X' / g^{yz}.H(A, S, g^x)^{pw1}$ and $g^{xyz} \leftarrow (g^{yz})^x$. Then, she checks whether $\alpha \leftarrow H(A, B, g^{xyz})$ holds or not. If it does not hold, A terminates the protocol. Otherwise, she is convinced that g^{xyz} is valid. And in this case, she can compute the session key $SK_A \leftarrow H'(A, B, g^{xyz})$ and returns $\beta \leftarrow H'(B, A, g^{xyz})$ to B for validation.

B3: Upon receiving β, B checks whether $\beta = H'(B, A, g^{xyz})$ holds or not. If it does hold, B can compute the session key $SK_B \leftarrow H'(A, B, g^{xyz})$. Otherwise, he terminates the protocol.

2.2 The undetectable password guessing attack on S-3PAKE and improved S-3PAKE.

In this section we briefly present the undetectable password guessing attack on S-3PAKE protocol and the improved version as reported by Gao et al., [11]. The malicious party B tries to guess the password of A by using the reply from the trusted server S.

Step 1:

A1: A chooses a random number $x \in Z_p$ and computes $X \leftarrow g^x N^{pw1}$, and then sends A||X to B.

B1: Upon receiving A||X guesses the password of A as pw1' and

computes $g^{x'} \leftarrow \frac{g^{x}.M^{pw1}}{M^{pw1'}}$. Then chooses a random number $y \in Z_p$ and computes $Y \leftarrow (g^{x'y})N^{pw2}$, then sends A||X ||B||Y to S and chooses randomly X′ and α′ and then sends to A. A would verify α.

Step2:

S2: : Upon receiving A||X ||B||Y , the server S first uses the passwords pw1 and pw2 to compute $g^{x} \leftarrow X/M^{pw1}$ and $g^{y} \leftarrow Y/N^{pw2}$, respectively. Then she chooses another random number $z \in Z_p$ and computes $g^{zx} \leftarrow (g^{x})^{z}$, $g^{x'yz} \leftarrow (g^{x'yz})^{z}$. Finally, she sends X′||Y′ to B, where $X' \leftarrow g^{x'yz}H(A,S,g^{x})^{pw1}$ and $Y' \leftarrow g^{xz}H(B,S,g^{x'y})^{pw2}$.

B2: When B receives $X'\|Y'$, he uses the password pw2 to compute $Y^{\sim} \leftarrow Y'/H(B,S,g^{x'y})^{pw2}$ and $X^{\sim} \leftarrow X'/H(A,S,g^{x'})^{pw1'}$ and checks $(g^{xz})^{y} \leftarrow g^{x'y^{z}}$, if this is correct , then pw1=pw1'.

2.3 Improved S-3PAKE

The keys K_{AS} and K_{BS} are established securely between A and S, B and S, respectively.

Step 1:

A1: A chooses a random number x $\in Z_p$ and computes X $\leftarrow g^{x}.N^{pw1}$ and δ_A = M AC_{KAS} (X), then sends A||X||δ_A to B, where M AC(X) is the message authentication code of X .

B1: B also chooses a random number y $\in Z_p$ and computes Y $\leftarrow g^{y}.N^{pw2}$ and δ_B = M AC_{KBS} (Y), then sends A||X ||$\delta_{A||}$Y||δ_B to S.

Step 2:

S2: Upon receiving A||X||δ_A||B||Y||δ_B , the server S first verify δ_A and δ_B , then uses the passwords pw1 and pw2 to compute $g^{x} \leftarrow X/M^{pw1}$ and $g^{y} \leftarrow Y/N^{pw2}$, respectively. Then she chooses another random number z $\in Z_p$ and computes $g^{xz} \leftarrow (g^{x})^{z}$, $g^{yz} \leftarrow (g^{y})^{z}$. Finally, she sends X'||Y' to B, where $X' \leftarrow g^{yz}.H(A, B, S, g^{x})^{pw1}$ and $Y' \leftarrow g^{xz}.H(B, A, S, g^{y})^{pw2}$.

B2: when B receives X' ||Y', he uses the password pw2 to compute $g^{xz} \leftarrow Y'/g^{xz}.H(B, A, S, g^{y})^{pw2}$, and uses random number y to compute $g^{xyz} \leftarrow (g^{xz})^{y}$.

At last, he forwards X′||α to A, where $\alpha \leftarrow H(A, B, g^{xyz})$.

Step 3:

A3: After A receives X'||α, she first computes $g^{yz} \leftarrow X'/g^{yz}.H(A, B, S, g^{x})^{pw1}$ and $g^{xyz} \leftarrow (g^{yz})^{x}$. Then, she checks whether $\alpha \leftarrow H(A, B, g^{xyz})$ holds or not. If it does not hold, A

terminates the protocol. Otherwise, she is convinced that g^{xyz} is valid. And in this case, she can compute the session key $SK_A \leftarrow H'(A, B, g^{xyz})$ and returns $\beta \leftarrow H'(B, A, g^{xyz})$ to B for validation.

B3: Upon receiving β, B checks whether $\beta = H'(B, A, g^{xyz})$ holds or not. If it does hold, B can compute the session key $SK_B \leftarrow H'(A, B, g^{xyz})$. Otherwise, he terminates the protocol.

3. Online Undetectable Password Guessing Attack

The improved S-3PAKE still suffers from undetected online attack, which causes the malicious party B to guess the password by using the reply from the server S.

Step 1:

A1: A chooses a random number $x \in Z_p$ and computes $X \leftarrow g^x.N^{pw1}$ and $\delta_A = MAC_{KAS}(X)$, then sends $A||X||\delta_A$ to B, where MAC(X) is a message authentication code of X.

B1: Upon receiving $A||X||\delta_A$, B chooses a random number y, guesses the password of A as pw1' and computes $g^{x'} \leftarrow \frac{g^x.M^{pw1}}{M^{pw1'}}$. Then chooses a random number $y \in Z_p$ and computes $Y \leftarrow (g^{x'y})N^{pw2}$, $\delta_B \leftarrow MAC_{BS}(Y)$ and sends $A||X||B||\delta_A||Y||\delta_B$ to S

Step2:

S2: : Upon receiving $A||X||B||Y||\delta_B$, the server S first verify $\delta_A \delta_B$, uses the passwords pw1 and pw2 to compute $g^x \leftarrow X/M^{pw1}$ and $g^{x'y} \leftarrow Y/N^{pw2}$, respectively. Then she chooses another random number $z \in Z_p$ and computes $g^{zx} \leftarrow (g^x)^z$, $g^{x'yz} \leftarrow (g^{x'yz})^z$. Finally, she sends $X'||Y'$ to B, where $X' \leftarrow g^{x'yz} H(A, B, S, g^x)^{pw1}$ and $Y' \leftarrow g^{xz} H(B, A, S, g^{x'y})^{pw2}$.

B2: When B receives X' ||Y' , he uses the password pw2 to compute $Y^{\sim} \leftarrow Y'/H(B, A, S, g^{x'y})^{pw2}$ and $X^{\sim} \leftarrow X'/H(A, B, S, g^{x'})^{pw1'}$ and checks $(g^{xz})^y \leftarrow g^{x'y^z}$, then pw1=pw1'.

3.1 Discussion and Countermeasures

This section presents a simple way of avoiding the above attack by using the ephemeral secret keys K_{AS} and K_{BS} between the server and the clients A and B for the computations of X' and Y' of the server.

The procedure is as follows:-

Step 2:

S2: Upon receiving $A||X||\delta_A||Y||\delta_B$, the server S first verify δ_A and δ_B, then uses the passwords pw1 and pw2 to compute $g^x \leftarrow X/M^{pw1}$ and $g^y \leftarrow Y/N^{pw2}$, respectively. Then she

chooses another random number $z \in Z_p$ and com-putes $g^{xz} \leftarrow (g^x)^z$, $g^{yz} \leftarrow (g^y)^z$. Finally, she sends $X' \| Y'$ to B, where $X' \leftarrow g^{yz} .g^{KAS} .H(A, B, S, g^x)^{pw1}$ and $Y' \leftarrow g^{xz} .g^{KBS} .H(B, A, S, g^y)^{pw2}$.

B2: when B receives $X' \| Y'$, he uses the password pw2 to compute $g^{xz} .g^{KBS} \leftarrow Y'/H(B, A, S, g^y)^{pw2}$, next $g^{xz} \leftarrow g^{xz} .g^{KBS} .(g^{KBS})^{-1}$ and uses random number y to compute $g^{xyz} \leftarrow (g^{xz})^y$. At last, he forwards $X' \| \alpha$ to A, where $\alpha \leftarrow H(A, B, g^{xyz})$.

Step 3:

A3: After A receives $X' \| \alpha$, she first computes $g^{yz} .g^{KBS} \leftarrow X'/H(A, B, S, g^x)^{pw1}$ and $g^{xyz} \leftarrow ((g^{yz}).g^{KAS} .(g^{KAS})^{-1})^x$. Then, she checks whether $\alpha \leftarrow H(A, B, g^{xyz})$ holds or not. If it does not hold, A terminates the protocol. Otherwise, she is convinced that g^{xyz} is valid. And in this case, she can compute the session key $SK_A \leftarrow H'(A, B, g^{xyz})$ and returns $\beta \leftarrow H(B, A, g^{xyz})$ to B for validation.

B3: Upon receiving β, B checks whether $\beta = H(B, A, g^{xyz})$ holds or not. If it does hold, B can compute the session key $SK_B \leftarrow H'(A, B, g^{xyz})$. Otherwise, he terminates the protocol.

4. Conclusion

In the present paper we demonstrated the undetectable password guessing atttack on improved S-3PAKE protocol. The message authentication code for X and Y is not sufficient to preclude the above attack. The efficient way to make the system to resist this attack is by including secret random ephemeral keys between A and S, S and B in the computations of X' and Y'. Apart from all these facts, the provable security approach should be taken when designing the protocols.

References

[1] Abdella, M. and Pointcheval, D, Simple password-based encrypted key excahnge protocols, CT-RSA 2005, Springer-Verlag, vol 3376, pp. 191-208, 2005.

[2] Bellovin, S.M. and Merrit, M, Encrypted key exchange: password- based protocols secure against dictionary attacks. In: Proceedings of IEEE sysmposium on research in security and privacy, IEEE Computer society press pp. 72-84, 1992.

[3] Chang, C.C. and Chang, Y.F, A novel three party encrypted key exchange protocol, Computer Standards and Interfaces, 26(5), pp. 471-6, 2004.

[4] Diffie, W. and Hellman, M, New Directions in cryptography, IEEE Transactions on Information theory, 22(6), pp. 644-54, 1976

[5] Ding, Y. and Hoster, P, Undetectable Online password guessing attacks, ACM operatinf system review, 29(4), pp. 77-86, 1995

[6] Guo, H., Li, Z., Mu, Y. and Zhang, X, Cryptanalysis of simple three-party key exchange protocol , Computers and Security, 27(1), pp. 16-21, 2008

[7] Lin, C.L., Sun, H.M., Steiner, M. and Hwang, T, Three-party excrypted key exchange without server public keys IEEE Communication letters, 5(12), pp. 497-9, 2001

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

[9] Phan, R.C.W., Yau, W.C. and Goi, B.M. Cryptanalysis of simple three-party key exchange protocol (S-3PAKE), Information sciences, 178(13), pp. 2849-2856. 2008.

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

[11] Yoon, E.J. and Yoo, K.Y, Improving the novel three-party encrypted key exchange protocol, Computer Standards and Interfaces, 30(5), pp. 309-314, 2008.