A Simple Three Party Password based Key Exchange Protocol

Tang,Wen
Dept of computer. Hunan Traditional Chinese Medical College
Hunan zhuzhou, China
e-mail: karontom-811103@hotmail.com

***Abstract*—The three-party password-based key exchange protocol (3PAKE) is the protocol in which two communications entities can authenticate each other and establish a session key through the assistance of an authentication server. All exsisting 3PAKE need certain encrypion system which are not suitable to some certain application such as low devices. In this paper, we propose a simple three-party password-based key exchange protocol(S3PAKE) without encrypton based on augmented password by this way that every client only shares a common password with a trusted server and any two clients can authenticate each other and negotiate a session key relying on the help of the trusted server. Compared to previous protocols, our proposed protocols are more efficent and convenient since the protocol need not encrypt passwords..**

Keywords password attack; password ; three party; protocol; authentication; key exchange

I. INTRODUCTION

Password authenticated key exchange protocols (PAKE) is the protocol that two parties who only a share a human-memorable password and who are communicating over an insecure network want to authenticate each other and negotiate a session key to be used for protecting their later communication. Designing a secure PAKE is non-trivial thing due to the fact that the password is picked up from a small space so that the protocol is vulnerable to dictionary attacks. Bellovin and Merritt [1] first proposed the encrypted key exchange which enables two communication entities authenticate each other and establish a shared session key. Since then, number two party passwords based authenticated key exchange (2PAKE) protocols [2-3, 9-11] have been proposed to improve security and performance. Although 2PAKE protocols are quite useful for client-server architectures, they are not suitable to large scale communication environments since 2PAKE protocols require every pair of communication entities to share a password it is every inconvenient in key management for client-client communications in large scale communication environments. To address this problem, some three party passwords based authenticated exchange protocols have been proposed (3PAKE) [4-7]. In a three-party PAKE, each client (user) first shares a human-memorable password with a trusted server (TS), and then when two users wish to create a session key, they resort to the TS for authenticating each other. With the sharing passwords, TS can help the two clients to authenticate each other and construct a common key for secure communication. In 1995, Steiner et al. [1] developed a 3PAKE protocol, which is based on Diffie-Hellman key exchange concept, to improve system efficiency by reducing the number of transmission rounds and cryptographic operations in comparison with Bellovin and Merrit's protocol. Later, Ding and Horster [5] pointed out that Steiner et al.'s protocol cannot resist the undetectable on-line password guessing attacks and developed an improved scheme to solve this security vulnerability. Lin et al. [6] also demonstrated a series of malicious procedures of invoking off-line password guessing attacks on Steiner et al.'s protocol. For eliminating these authentication flaws, Lin et al. utilized public key cryptographic technology to construct their remedy and achieve security enhancement. However, the public key technology is still too expensive to be afforded in current 3PAKE protocol. In order to achieve high performance, in [7], the authors proposed a 3PAKE without using public key cryptographic, but the proposed also needs to use the password as the authentication by encrypting the password.

In this paper, we proposed a simple 3PAKE by introducing the idea of [8]. Compared to previous 3PAKE, our protocol is more efficient and convenient since the protocol need not encrypt passwords.

II. THE PROPOSED PROTOCOL

In this section, we present a new simple 3PAKE protocol. First we introduce some notations used in our paper. First, the system chooses two large prime number p, q, and find an integer g, which is a primitive element in both GF (p) and GF (q). In order illustrate the protocol clearly; some notations are introduced as follows:

p, q: denote two large prime number.
g: a primitive element in both GF(p) and GF(q)
TS: a trust sever
A , B : the initiator and the responder, respectively
pw_A : the password shared between user A and TS
pw_B : the password shared between user B and TS
$h()$: a public one-way hash function.
ID_A : the identity of A
ID_B : the identity of B
$||$: the concatenation operation

978-1-4244-8102-6/10/$26.00 © 2010 IEEE

$\oplus$: the bitwise XOR operation

Since the reason of suffering from undetectable on-line dictionary attack in [7] is that there is no mechanism for a trusted server to make sure whether the client is a valid user or nor. In order to resist to such an attack, we introduce the authentication for TS to a client. In our protocol, each client shares a human-memorable password with a trusted server (TS). When two clients want to establish a shared session key, they resort to the trusted server for authenticating each other. The details will be described in the following steps

Setp1: User A chooses a random number x and sends (ID_A, g^x) to B.

Setp2: User B **also** chooses a random number y and sends (ID_A, g^x, ID_B, g^y) to TS.

Setp3: Upon **receiving** (ID_A, g^x, ID_B, g^y), the TS chooses two random number a and b respectively, and then it uses the shared password pw_A and pw_B to compute $Z_A = g^{(x+pw_A)a}$, $Z_B = g^{(y+pw_B)b}$, respectively, Later, the TS sends (Z_A, Z_B) to A, B, respectively.

Step4: Once **receiving** Z_A from TS, A computes $g^a = g^{(x+pw_A)a(x+pw_A)^{-1}}$, and then the user A computes $K_A = h(g^a)$. Later, A sends K_A to TS for authenticating itself to TS.

Step5: Upon receiving Z_B, B also computes $g^b = g^{(x+pw_B)b(x+pw_B)^{-1}}$, and then the user B computes $K_B = h(g^b)$. Later, B sends K_B to TS for authenticating itself to TS.

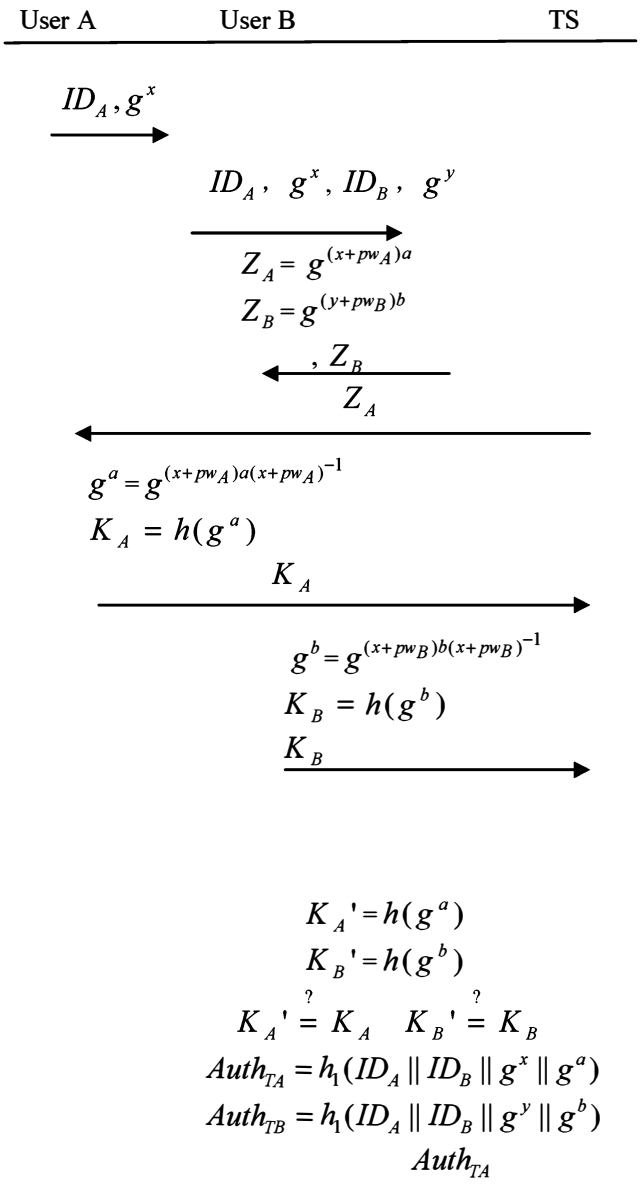
Step6: When receiving K_A and K_B, the TS computes

$K_A' = h(g^a)$ and $K_B' = h(g^b)$ and then, it checks whether K_A is equal to K_A' and K_B is equal to K_B', respectively. If not, S stops the session because the TS believe that there the user is not a valid one. Otherwise, TS computes $Auth_{TA} = h_1(ID_A \| ID_B \| g^x \| g^a)$ and $Auth_{TB} = h_1(ID_A \| ID_B \| g^y \| g^b)$. Later, TS sends $Auth_{TA}$ and $Auth_{TB}$ to B for authenticating itself to A and B, respectively.

Step7: After receiving ($Auth_{TA}$ $Auth_{TB}$) from TS, the B computes $Auth_{TB}' = h_1(ID_A \| ID_B \| g^y \| g^b)$ and checks whether $Auth_{TB}'$ is equal to $Auth_{TB}$. If yes, B computes session $K = g^{xy}$ and $Auth_{BA} = h_2(g^{xy})$, and then it sends $Auth_{BA}$ and $Auth_{TA}$ to A ; Otherwise, B stop the protocols.

Step8. Upon receiving ($Auth_{TA}$ $Auth_{BA}$), A checks whether the $Auth_{TA}$ is the wanted message. If yes, that indicates the TS is a valid trusted server and then also checks whether $Auth_{BA}$ is the wanted message. If yes, that indicates the B also is the valid user. Till to now, A and B negotiate a shared session $K = g^{xy}$, which is used for protect their later communication. Later, A sends $Auth_{AB} = h_2(g^{xy})$ to B for authenticating itself to B

Step9: Upon receiving $Auth_{AB}$ the B validate the value is the wanted one. If yes, the K is a valid session key. Figure 1 shows the whole key exchange process:



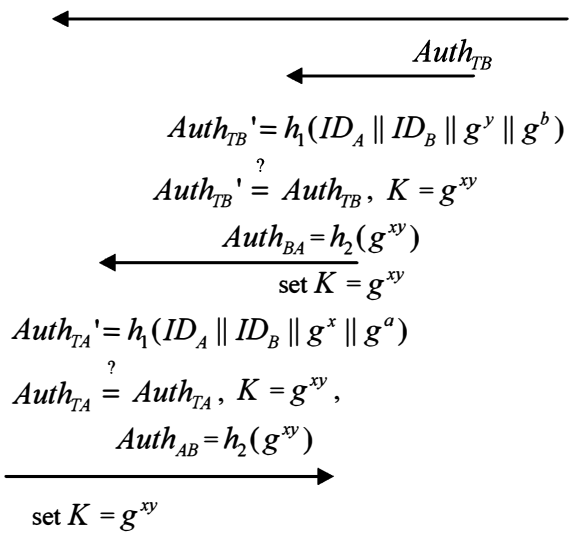


Figure 1. A simple Three Party Password based Key Exchange Protocols

III. SECURITY ANALYSIS

In this section, we will analyze that our proposed protocol is secure. Here we mainly discuss whether our proposed S3PAKE protocol can resist various known attacks.

1) On-line guessing attack

In our scheme, only the two communication entities are both legal users, the TS continues the protocol. Otherwise, the protocol terminates. In step 6, if the authenticated message is invalid, the TS will be aware of whose password has been targeted as a dictionary attacks. So the proposed protocol can resist to undetected online guessing attack.

2) Off-line guessing attack

In an off-line guessing attack, an attacker guesses a password and confirms his guess off-line. However, there is also no useful information to help verify the correctness of the guessed passwords in our protocol. Therefore, our protocol can resist the off-line guessing attack.

3) Man-in-the-middle attack

In our protocols, the adversary can not launch man-in the middle attack because the adversary can not generate K_A or K_B without knowing user's password. That is, it can not authenticate itself to TS.

IV. CONCLUSION

Three-party password-based authenticated key exchange (3PAKE) protocols are important cryptographic techniques for secure communications. Conceptually, a typical three-party password-based authenticated key exchange protocol works as follows. As requirement, each client shares a human-memorable password with a trusted server so that they can resort to the trusted server for authentication when both of them want to establish a shared session key. In this paper, we proposed a simple 3PAKE.Compared to previous 3PAKE; our protocol is more efficient and convenient since the protocol need not encrypt passwords.

REFERENCES

[1] Bellovin S M, Merritt M. Encrypted key exchange: Pass word-based protocols secure against dictionary attacks. Proceedings of the 1992 IEEE Computer Society Symposium on Research in security and Privacy. Oakland , California ,USA , 1992 : 72-84

[2] D.Jablon. Strong Password-Only Authenticated Key Exchange. ACM Computer Communications Review, October 1996.

[3] S.M.Bellovin and M.erritt. Encrypted: Key Exchange: Password-Based Protocols against Dictionary attacks. Proceeding of the IEEE Symposium on Research in Security and Privacy. Oakland, May 1992.

[4] M.Steiner, G. Tsudik, M.Waidner. Refinement and Extension of Encrypted Key Exchange [J]. ACM Operation Systems Review, 1995,29(3):22-30

[5] Y. Ding, P. Horster, Undetectable on-line password guessing attacks, ACM Operating Systems Review 29 (4) (1995) 77–86.

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

[7] C.L. Lin, H.M. Sun , M.S,et al. Three party Encrypted Key Exchange without Server Public Keys [J]. IEEE Commnuications Letters, 2001, 5(12):497-499

[8] T. Kwon. Authentication and Key Agreement via Memorable Password [EB/OL]. Internet Society Network and Distributed System Security Symposium, 2001. http://eprint. tacr.org/2000/026, 2004.5.10

[9] D. Jablon, Strong password-only authenticated key exchange, ACM Computer Communications Review, October 1996

[10] V. Boyko, P. MacKenzie, and S. Patel, "Provably Secure Password Authenticated Key Exchange Using Diffie-Hellman", Advances in Cryptology - EUROCRYPT 2000.

[11] S. Bellovin and M. Merritt, "Augmented Encrypted Key Exchange: a Password-Based Protocol Secure Against Dictionary Attacks and Password File Compromise", ATT Labs Technical Report, 1994