Threshold Password-Based Authentication Using Bilinear Pairings

Songwon Lee JeongKyu Yang Kwangjo Kim

International Research center for Information Security(IRIS),
Information and Communications Univ.(ICU),
58-4 Hwaam-Dong, Yuseoung-Gu, Daejeon, Korea 305-732

{swonlee, speed, kkj}@icu.ac.kr

Abstract We present a new threshold password-based authentication protocol that allows a roaming user(a user who accesses a network from different client terminals) to download a private key from remote servers with knowledge of only his identity and password. He does not carry the smart card storing user's private information. We note that as a goal of a multi-server roaming system, a protocol has to allow a user to get his private key from the servers, even if some of the servers are compromised. With this point of view, we give the first description of a threshold password-only roaming protocol. In this paper, we use (k,n)-*threshold scheme* in which only k honest servers or more are engaged to reconstruct a secret key. Our scheme is based on bilinear pairings which could be built from Weil pairing or Tate pairing.

1 Introduction

With rapid development on the Internet a user Bob can easily access the network to get some services from a service provider, or to retrieve his sensitive private data stored in the server previously. In that case, he has to convince the server that he is a really legitimate user. To verify the identity(ID for short) of a user many real systems use password-based authentication. The fundamental problems with passwords come from the fact that most users' passwords are drawn from a relatively small spaces and are easily memorable, which also means that the password may be easily guessed by an attacker.

Let us assume that a roaming user accesses a network from different client terminals to download a private key from remote servers with knowledge of only his ID and password. He does not carry the smart card storing user's private information. While the smart card plays an important role in storing sensitive information, it is impractical in many real environments due mainly to inconvenience, for example, a user needs an external interface device to communicate with a smart card. With mainly focused on this point, strong password authentication protocols were presented by Perlman *et al.*[13], Ford *et al.*[6], and Jablon[9], *etc.* Some of them used multiple servers to implement a roaming protocol that uses a weak secret key to securely retrieve and reconstruct a strong private key that has been divided into pieces distributed among multiple servers. We note that as one of goals, a protocol has to allow a user to get his private key from the servers, even if some of the servers are compromised.

In this paper we present a threshold password based authentication protocol for a roaming user. We make use of the (k,n)-*threshold scheme* in which only k honest servers or more are engaged to reconstruct a secret key. Our scheme is based on bilinear pairings that could be built from Weil pairing or Tate pairing over *Gap Diffie-Hellman(GDH)* group, which *Computational Diffie-Hellman(CDH)* problem is hard but *Decision Diffie-Hellman(DDH)* problem is easy.

2 Preliminaries

2.1 Previous Works

Perlman and Kaufman presented protocols [13] that one can securely retrieve a private key and use this to download the rest of one's security context. Ford and Kaliski proposed methods [6] that use multiple servers to prevent attacking by introducing *password hardening protocol* by which servers interact with the user to harden the user's password into a strong secret without revealing either the user's password or the hardened result. A *password-only multi-server roaming protocol* [9] is presented by Jablon. In his protocol, the user can authenticate servers and retrieve his private key for temporary use on a client machine using just an easily memorable password. [6] and [9] make use of the multiple servers to gain the goal of the protocol. When some of the servers are compromised, the user can not obtain valid secret key no matter what the user has a method to verify the key. In our proposed scheme, we mainly address this problem.

Let's briefly review here the protocol proposed in [9]. In this protocol the author introduced *forgiveness protocol* by which user's honest mistakes are forgiven by sending evidence of recent prior invalid access attempts after each successful authentication. But we do not consider here this forgiveness.

Parameters. The protocol operates in a subgroup of order q in $\mathbb{Z}_p^*$,where p, q and r are odd primes, $p = 2rq + 1, 2^{k-1} < p < 2^k, r \neq q$, and $2^{2_j-1} < q < 2^{2_j}$.

Enrollment. The user, Alice, selects a password π, computes $g_\pi = h(\pi)^{2r}$, and creates a private key U. For each $i \in [1, n]$, she computes a secret key share $S_i = {g_\pi}^{y_i}$ using randomly chosen $y_i \in_{\mathcal{R}} [1, q-1]$. She then creates her master j-bit symmetric key with $K_m = h(S_1 \parallel \cdots \parallel S_n) \bmod 2^j$, creates her encrypted private key as $U_K =_{K_m} \{U\}$, and then creates her key verifier $proof_{PK_m} = h(K_m \parallel g)$.

1. *Client*: send $\{ID_A, y_i, U_K, proof_{PK_m}\}$ to each server L_i for all $i \in [1, n]$.
2. *Servers*: store them in a list C_i maintained on each server.

Authenticated Retrieval. To retrieve her master key at a later time, the client and servers perform the protocol as below:

1. *Client*: select a random number $x \in [1, q-1]$, computes $X = {g_\pi}^x \bmod p$, and then send $\{ID_A, X\}$ to *Servers*.
2. *Servers*: retrieve $\{ID_A, y_i, U_K, proof_{PK_m}\}$ from C_i, compute $Y_i = X^{y_i}$, and then reply $\{Y_i, U_K, proof_{PK_m}\}$ to *Client*.
3. *Client*: compute $S_i = Y_i^{1/x} \bmod p$ for each $i \in [1, n]$, and then generate $K' = h(S_1 \parallel S_2 \parallel \cdots \parallel S_n)$. If $proof_{PK_m} \neq h(K' \parallel g)$ abort, otherwise compute $U =_{1/K'} \{U_K\}$.

2.2 Threshold Cryptosystem

The concept of a threshold scheme, called secret sharing scheme was introduced in [14] and since then many researchers have investigated such schemes.

A (*k*,*n*)-*threshold secret sharing scheme* is a protocol among n players in which the *dealer* distributes partial information (*share*) about a *secret* to n participant such that:

- Any group of fewer k participants can not obtain any information about the secret.
- Any group of at least k participants can compute the secret in polynomial time.

In the next section, we describe our proposed password-based authentication scheme making use of the (*k*,*n*)-*threshold scheme* in which a user distributes secrets to multiple servers, assuming $n \geq 2k - 1$ [12, 10].

2.3 Bilinear Pairings

Let us consider an additive group $\mathbb{G}_1$ and a multiplicative group $\mathbb{G}_2$ of the same order q. Assume that the discrete logarithm problem is hard in both groups. Let P be a generator of $\mathbb{G}_1$, and $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ a bilinear map satisfying the following properties:

1. *Bilinearity*: $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$ for all $P, Q \in \mathbb{G}_1$ and all $a, b \in \mathbb{Z}$.
2. *Non-degeneracy*: if $\hat{e}(P, Q) = 1$ for all $Q \in \mathbb{G}_1$, then $P = \mathcal{O}$.
3. *Computability*: there exists an efficient algorithm to compute $\hat{e}(P, Q)$ for any $P, Q \in \mathbb{G}_1$.

With such groups $\mathbb{G}_1$ and $\mathbb{G}_2$, we can define the following hard cryptographic problems:

- Discrete Logarithm(DL) problem: Given $P, P' \in \mathbb{G}_1$, find an integer n such that $P = nP'$ whenever such integer exists.
- Computational Diffie-Hellman(CDH) problem: Given $(P, aP, bP) \in \mathbb{G}_1$ for $a, b \in \mathbb{Z}_q^*$, compute abP.
- Decision Diffie-Hellman(DDH) problem: Given $(P, aP, bP, cP) \in \mathbb{G}_1$ for $a, b, c \in \mathbb{Z}_q^*$, decide if $c = ab \pmod q$ or not.
- Gap Diffie-Hellman(GDH) problem: A class of problems where the CDH problem is hard but DDH problem is easy.

To construct the bilinear pairing, we can use the Weil pairing and Tate pairing. $\mathbb{G}_1$ is a cyclic subgroup of the additive group of points of a supersingular elliptic curve $E(\mathbb{F}_p)$ over a finite field while $\mathbb{G}_2$ is a cyclic subgroup of the multiplicative group associated to a finite extension of $\mathbb{F}_p$.

3 Proposed Scheme

We now present our enhanced model for a password-only threshold roaming protocol followed by the detailed description of our protocol.

3.1 The Model

The model we propose is similar to that of [9], but with some different features.

First, our scheme employs the concept of threshold, which permits the user playing the role of a dealer to distribute secret shares to n servers. But, we use the protocol in [14] in a different way so that only the user can obtain the secret value in collaboration with threshold servers, but no server can.

Second, the scheme is based on a ID-based cryptosystem such as IBE scheme [2] presented by Boneh and Franklin, assuming that there is a thrusted authority(TA for short) which generates a private key for a user.

On the other hand, although we don't consider *forgiveness protocol* introduced in [9], this protocol can be used in our scheme.

Enrollment. The user Bob constructs (k,n)-threshold system in a similar way as in [14].

We assume here that Bob has chosen his own identity as a public key Q_{ID} and obtained the corresponding private key D_{ID} generated by TA.

Bob then creates his encrypted private key D_K using a master key K_m. Finally, he creates a proof value V that links the password to his master key.

Bob sends secretly share Y_i, encrypted private key D_K, and the proof value V to each the n servers.

Authenticated Retrieval. When Bob wants to login the server at any available client terminal, he first performs the threshold protocol with at least k servers to recover his secret value in a way that uses the *Lagrange Interpolation* such as in [14].

Note here that no client terminal has Bob's information created at enrollment time.

Bob randomly chooses k servers and sends them a randomly blinded request message. On receiving the request, each server in turn responds with a blinded reply. At least one of the servers also sends Bob his encrypted private key D_K and proof value V.

Bob unblinds each reply to recover the se-

cret value and reconstructs his master key K_m using the secret value and password, and then verifies that the master key is correct using the proof value V and his password π. Finally, if the master key is correct, Bob gets his private key.

3.2 Our Protocol

We now describe our proposed scheme in detail.

Setup. The readers can refer to [6] for a detailed description of the ID-Based Encryption scheme by Boneh and Franklin. The TA playing the role of the Private Key Generator in [6] chooses two group $\mathbb{G}_1$ being GDH group and $\mathbb{G}_2$ of the same prime order q, a bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$, a generator $P \in \mathbb{G}_1$, a secret master key $s \in_R \mathbb{Z}_q^*$. The TA then sets $P_{pub} = sP$, chooses cryptographic hash functions $H_1 : \{0,1\}^* \rightarrow \mathbb{G}_1^*$ and $H_2 : \mathbb{G}_2 \rightarrow \{0,1\}^n$ for some n.

The public parameters are

$$\textbf{params}=\{\mathbb{G}_1, \mathbb{G}_2, \hat{e}, H_1, H_2, P, P_{pub}\}.$$

The user Bob is given his private key D_{ID} to be $D_{ID} = sQ_{ID}$ where $Q_{ID} = H_1(ID)$.

Enrollment. Bob picks a password π, computes $R_{ID} = H_1(\pi)$, and chooses a polynomial of degree $k-1$

$$f(x) = a_0 + a_1 x + \cdots + a_{k-1}x^{k-1}$$

for random $a_0, \ldots, a_{k-1} \in \mathbb{Z}_q^*$ where a_0 is his secret value. Bob now acts as the following to enroll his credentials.

1. Compute $y_i = f(i)$ and $Y_i = y_i Q_{ID}$ which is n key shares.
2. Create his master symmetric key with $K_m = H_2(\hat{e}(R_{ID}, Q_{ID})^{a_0})$, and then create his encrypted private key $D_K =_{K_m} \{D_{ID}\}$ and key verifier $V = H_2(K_m \parallel P_{pub})$.

The client sends $\{Q_{ID}, Y_i, D_K, V\}$ to each server L_i for all $i \in [1, n]$. On receiving, servers store them.

Authenticated Retrieval. For authenticated retrieval, the client sends k servers a request message as the following. Denote k servers by $S = \{L_i \mid 1 \leq i \leq k\}$.

1. Select a random number $x \in [1, q-1]$, compute $X = xR_{ID}$.
2. Send X and Q_{ID} to every server L_i for $i \in S$.

On receiving the request, each server responds as follows:

1. Retrieve $\{Q_{ID}, Y_i, D_K, V\}$ from the storage maintained securely on each L_i.
2. Compute $R_i = \hat{e}(X, Y_i)$, and then reply $\{R_i, D_K, V\}$ to the client.

Finally, the client reconstructs Bob's private key by performing the following:

1. Compute $l_i = \prod_{j \in S, j \neq i} \frac{j}{j-i}$ for each i^{th} server.
2. Compute $R_i' = R_i^{l_i x^{-1}}$ and $K' = \prod_{i \in S} R_i'$.
3. Generate $K_m' = H_2(K')$.
4. If $V \neq H_2(K_m' \parallel P_{pub})$, abort.
5. To obtain the private key, decrypt D_K with the master key K_m'.

The correctness of the scheme is easy to verify since

$$\begin{aligned}
K' &= \prod_{i \in S} \hat{e}(xR_{ID}, y_i Q_{ID})^{l_i x^{-1}} \\
&= \prod_{i \in S} \hat{e}(y_i l_i R_{ID}, Q_{ID}) \\
&= \prod_{i \in S} \hat{e}(f(i) \prod_{j \in S, j \neq i} \frac{j}{j-i} R_{ID}, Q_{ID}) \\
&= \hat{e}(\sum_{i=1}^{k} f(i) \prod_{j \in S, j \neq i} \frac{j}{j-i} R_{ID}, Q_{ID}) \\
&= \hat{e}(a_0 R_{ID}, Q_{ID}).
\end{aligned}$$

4 Security Analysis

In this section we now roughly discuss the security aspects of our proposed scheme. We basically consider two kinds of attacks and robustness of the scheme.

First, we can assume that an attacker has no knowledge of R_{ID}, that is $H_1(\pi)$. The protocols must not provide enough information to prevent the attacker from performing a dictionary attack against π or R_{ID}. Second, an attacker that knows R_{ID} should neither be able to pretend the user to the server, nor be able to learn useful information about π (without running a dictionary attack) [4].

With respect to robustness, the user is able to obtain a valid private key without revealing both π and any related private information.

Attacks on either R_{ID} and π. We assume that some (fewer than k) of the servers are compromised and an adversary obtains R_{ID}. We now are interested in protecting π and the master key K_m. None of the information about π except R_{ID} is transferred during the communications. In that case, no adversary can obtain π and even K_m under the assumption that the *Discrete Logarithm Problem (DLP)* and *Computational Diffie-Hellman Problem (CDHP)* are hard.

Furthermore, even if the user completes the protocol unsuccessfully by compromising more than k servers, the master secrete key K_m is not revealed.

Robustness[15, 12]. The robustness ensures that the scheme can be resistant to corruption of even $k-1$ of $n \geq 2k-1$ servers. A user chooses randomly a secret value a_0 uniformly distributed in $\mathbb{Z}_q^*$ during enrollment to construct threshold scheme.

Hence even there exists an adversary who be able to corrupt at most $k-1$ servers among $n \geq 2k-1$, any subset of k servers constructs the secret value uniformly distributed in $\mathbb{Z}_q^*$. No corrupted server thus can get information about the secret value.

We leave ourselves a rigorous security analysis for our scheme as a further work.

5 Comparison

With mainly compared to [6] and [9], our scheme is capable of resisting that fewer than threshold servers are compromised. When only k honest servers are involved in the protocol, the user can retrieve his private key. However, both schemes mentioned above require none of the servers have compromised.

6 Conclusions

We present a new threshold password-only roaming protocol. It allows a roaming user to download a private key from remote servers, without revealing the password to off-line guessing. No client terminal has user's information created at enrollment time.

We note that, as a goal of a multi-server roaming system, a protocol has to allow a user to get his private key from the servers, even if some of the servers are compromised. With this point of view, we give the first description of a threshold password-only roaming protocol. In this paper, we use (k,n)-*threshold scheme* in which only k honest servers or more are engaged to reconstruct a secret key. Our scheme is based on bilinear pairings that could be built from Weil pairing or Tate pairing.

The protocol is useful for authenticating roaming users and even non-roaming users, and retrieving private keys for use in other applications.

References

[1] S.Al-Riyami and K.Paterson, "Certificateless Public Key Cryptography", available at $http:\backslash\backslash www.ime.usp.br/\sim rt/cranalysis/CertifLessPKC.pdf$, Jul.2003.

[2] D.Boneh and M.Franklin, "Identity-Based Encryption from the Weil

Pairing", *CRYPTO2001*, LNCS 2139, pp.213-229, Springer-Verlag, 2001.

[3] S.Bellovin and M.Merritt, "Encrypted Key Exchange: Password-based Protocols Secure against Dictionary Attacks", *Proc.IEEE Symposium on Research in Security and Privacy*, May 1992.

[4] S.Bellovin and M.Merritt, "Augmented Encrypted Key Exchange: A Password-based Protocol Secure Against Dictionary Attacks and Password File Compromise", Technical Report, AT&T Bell Laboratories, 1994.

[5] J.Baek and Y.Zheng, "Identity-Based Threshold Decryption", *IACR eprint*, 2003/164.

[6] W.Ford and B.Kaliski, "Server-Assisted Generation of a Strong Secret from a Password", *Proc.*9^{th} *International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprise*, IEEE, Jun.14-16, 2000.

[7] F.Hess, "Efficient Identity Based Signature Schemes Based on Pairings", *SAC2002*, LNCS 2595, pp.310-324, Springer-Verlag, 2003.

[8] D.Jablon, "Strong Password-Only Authenticated Key Exchange", *ACM Computer Communications Review*, Oct.1996.

[9] D.Jablon, "Password Authentication Using Multiple Servers", *CT-RSA2001*, LNCS 2020, pp.344-360, Springer-Verlag, 2001.

[10] B.Libert and J.Quisquater, "Efficient revocation and threshold pairing based cryptosystems", *PODC'03*, pp.163-171, Jul.13-16, 2003.

[11] P.MacKenzie, T.Shirmpton and M.Jakobsson, "Threshold Password-Authenticated Key Exchange(Extended Abstract)", *CRYPTO2002*, LNCS 2442, pp.385-400, Springer-Verlag, 2002.

[12] T.Pedersen, "Non-interactive and Information theoretic Secure Verifiable Secret Sharing", *CRYPTO'91*, LNCS 576, pp.129-140, Springer-Verlag, 1992.

[13] R.Perlman and C.Kaufman, "Secure Password-Based Protocol for Downing a Private Key", *Proc. 1999 Network and Distributed System Security Symposium*, Internet Society, Jan.1999.

[14] A.Shamir, "How to Share a Secret", *Communication of the ACM*, Vol.22, No.11, pp.612-613, Nov.1979.

[15] D.Vo, F.Zhang and K.Kim "A New Threshold Blind Signature Scheme from Pairings", *SCIS2003*, Vol.1/2, pp.233-238, Jan.2003.

[16] T.Wu, "The Secure Remote Password Protocol", *Proc. 1998 Network and Distributed System Security Symposium*, pp.97-111, Internet Society, Jan.1998.