A New Authentication Scheme For Session Initiation Protocol

Eun-Jun Yoon
School of Electrical Engineering and Computer Science
Kyungpook National University
Daegu 702-701, Republic of Korea
ejyoon@tpic.ac.kr

Kee-Young Yoo*
Department of Computer Engineering
Kyungpook National University
Daegu 702-701, Republic of Korea
yook@knu.ac.kr

Abstract

In 2008, Tsai proposed an efficient nonce-based authentication scheme for Session Initiation Protocol (SIP). The current paper, however, demonstrates that Tsai's authentication scheme is still vulnerable to off-line password guessing attacks, Denning-Sacco attack and stolen-verifier attacks, and does not provide perfect forward secrecy. We also propose a new secure and efficient authentication scheme based on the elliptic curve discrete logarithm problem (ECDLP) for SIP in order to overcome such security problems.

Keyword: Network security, Cryptography, Cryptanalysis, Authentication, Session initiation protocol

1 Introduction

In 1999, Internet Engineering Task Force (IETF) proposed the Session Initiation Protocol (SIP) for the IP-based telephony protocol [1][2][3][4][5]. Because SIP is a text-based peer-to-peer protocol, it uses Internet protocols such as Hyper Text Transport Protocol (HTTP) and Simple Mail Transport Protocol (SMTP) [6]. In 2005, Yang et al. [7] pointed out that the procedure of the original SIP authentication scheme based on HTTP digest authentication is vulnerable to the off-line password guessing attack and the server spoofing attack. They also proposed a secure authentication scheme for SIP to resist the attacks. Yang et al.'s scheme is based on Diffie-Hellman key exchange algorithm [8], which depends on the difficulty of Discrete Logarithm Problem (DLP). However, Yang et al.'s scheme does not suitable for low computation power equipments because the computation cost of the scheme is very high. Based on Yang et al.'s scheme, Durlanik et al. [9] proposed an efficient authentication scheme for SIP by using Elliptic Curve Diffie-Hemmman (ECDH) key exchange algorithm [10][11] in 2005. Durlanik et al.'s scheme can reduce the total execution times and the memory requirements in comparison with Yang et al.'s scheme by adoption elliptic-based key exchange mechanism.

In 2008, Tsai [12] also proposes an efficient authentication scheme based on the random nonce. Tsai's scheme is based on the random nonce. Since all communication messages are encrypted/decrypted by using one-way hash function and exclusive-or operation, the computation cost of Tsai's scheme is very low and it is very suitable for low computation equipment. Nevertheless, Tsai's scheme is still vulnerable to off-line password guessing attacks, Denning-Sacco attack and stolen-verifier attacks, and does not provide perfect forward secrecy [13][14][15]. Accordingly, the current paper demonstrates the vulnerability of Tsai's scheme to the attacks, and then proposes a secure and efficient authentication scheme based on the elliptic curve discrete logarithm problem (ECDLP) for SIP in order to overcome such security problems. The Elliptic Curve Cryptosystem (ECC) [10][11] presents an attractive alternative cryptosystem because its security is based on the elliptic curve discrete logarithm problem (ECDLP). ECC operates over a group of points on an elliptic curve and offers a level of security comparable to classical cryptosystems that use much larger key sizes. As a result, the proposed authentication scheme resists those attacks, while also providing more security and efficiency which can be executed faster than other previously proposed related schemes including the Tasi's scheme.

2 Review of Tsai's authentication scheme

This section briefly reviews Tsai's nonce-based authentication scheme for session initiation protocol [12]. Notations used in this paper are defined as follows:

- U: the remote user;
- S: the remote server;
- D: a uniformly distributed dictionary of size $|D|$;

*Corresponding author: Kee-Young Yoo (e-mail: yook@knu.ac.kr) Tel.: +82-53-950-5553; Fax: +82-53-957-4846

978-0-7695-3575-3/09 $25.00 © 2009 IEEE
DOI 10.1109/CISIS.2009.93

- PW: a low-entropy password of U chosen from D;
- EK_s: a high-entropy secret key of S;
- N: a random nonce generated by U and S;
- SK: a shared common session key between U and S;
- $X \rightarrow Y : M$: X sends a message M to Y;
- $h(\cdot)$: a secure one-way hash function;
- $\oplus$: a bit-wise exclusive-or(XOR) operation;
- $||$: a concatenation operation;

There are two phases in the Tsai's scheme: registration and authentication. Fig.1 illustrates Tsai's scheme and it proceeds as follows:

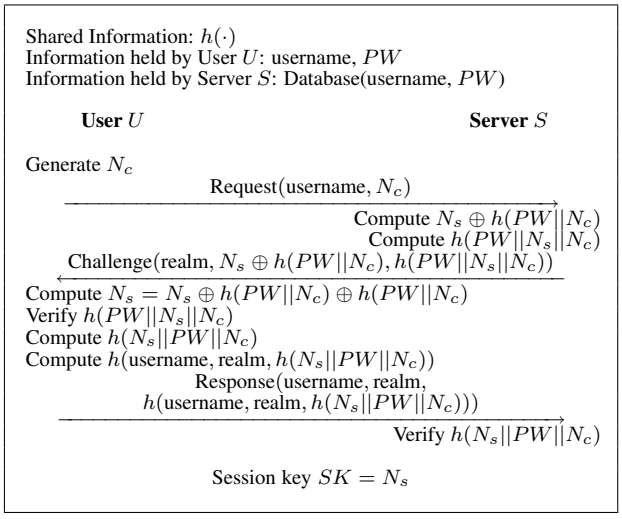

Figure 1. Tsai's authentication scheme

2.1 Registration phase

When a user U wants to register and become a new legal user, U and S execute the following steps:

1. $U \rightarrow S$: username, PW
   U submits his/her username and password PW to the remote server S.
2. S stores the U's username and PW in the user account database.

2.2 Authentication phase

If a legal user U wants to login into the remote server S, he/she first inputs his/her username and password PW into the client system. Then, the authentication phase proceeds as follows:

1. $U \rightarrow S$: Request(username, N_c)
   U generates a random nonce N_c and then sends it with a request message as Request(username, N_c) to S.
2. $S \rightarrow U$: Challenge(realm, $N_s \oplus h(PW||N_c), h(PW||N_s||N_c)$)
   Upon receiving the request message, S generates a random nonce N_s and then computes $N_s \oplus h(PW||N_c)$ and $h(PW||N_s||N_c)$. Finally, S sends a challenge message Challenge(realm, $N_s \oplus h(PW||N_c), h(PW||N_s||N_c)$) to U.
3. $U \rightarrow S$: Response(username, realm, h(username, realm, $h(N_s||PW||N_c)$))
   Upon receiving the challenge message, U computes $h(PW||N_c)$ and derives N_s by computing $N_s \oplus h(PW||N_c) \oplus h(PW||N_c)$. Then, U computes $h(PW||N_s||N_c)$ and verifies whether it is equal to the received challenge $h(PW||N_s||N_c)$. If they are not equal, U rejects the server challenge message. Otherwise, U authenticates S and computes two hash values $h(N_s||PW||N_c)$ and h(username, realm, $h(N_s||PW||N_c)$). Finally, U sends a response message Response(username, realm, h(username, realm, $h(N_s||PW||N_c)$)) to S.
4. Upon receiving the response message, S computes $h(N_s||PW||N_c)$ and verifies whether it is equal to the received response $h(N_s||PW||N_c)$. If they are not equal, S rejects the user response message. Otherwise, S authenticates U and accepts the user's login request.

After mutual authentication between U and S, $SK = N_s$ is used as a session key.

3 Cryptanalysis of Tsai's scheme

This section shows that Tsai's authentication scheme for session initiation protocol [12] is vulnerable to off-line password guessing attacks, Denning-Sacco attack and stolen-verifier attacks, and does not provide perfect forward secrecy [13][14].

3.1 Off-line password guessing attacks

Let Eve be an active attacker who interposes the communication between U and S. Then, Eve can easily obtain a legitimate communication parties' password PW. The off-line password guessing attacks proceed as follows:

1. When U sends Request(username, N_c) to S, Eve intercepts it.
2. When S sends Challenge(realm, $N_s \oplus h(PW||N_c)$, $h(PW||N_s||N_c)$) to U, Eve intercepts it.
3. In order to obtain the password PW shared between U and S, Eve makes a guess at the secret password PW^* from dictionary D.
4. By using the captured nonce N_c and guessed PW^*, Eve computes $h(PW^*||N_c)$ and derives a nonce N_s^* by computing $N_s^* = N_s \oplus h(PW||N_c) \oplus h(PW^*||N_c)$, where $N_s \oplus h(PW||N_c)$ is the information that Eve captured.

5. Eve checks if $h(PW||N_s||N_c) \stackrel{?}{=} h(PW^*||N_s^*||N_c)$, where $h(PW||N_s||N_c)$ is the information that Eve captured. If it holds, Eve has guessed the correct secret password $PW^* = PW$.
6. If it is not correct, Eve repeatedly performs above steps 3$\sim$5 until $h(PW||N_s||N_c) = h(PW^*||N_s^*||N_c)$.

3.2 Denning-Sacco attack

The Dennig-Sacco attack is where U or S compromises an old session key and an attacker tries to find a long-term private key (e.g. user password or server private key) or other session keys. This attack arises from the fact that the compromise of a fresh session key enables the protocol to be compromised. Such attacks have long been known. Please refer the Denning-Sacco attack in [14].

In Tsai's authentication scheme, suppose Eve has a session key $SK = N_s$ of the protocol. Then, knowledge of $SK = N_s$ will enable $h(PW||N_c)$ to be discovered from $N_s \oplus h(PW||N_c)$ by computing $N_s \oplus h(PW||N_c) \oplus SK$. Then, since N_c is open nonce value, the long-term private password PW included in $h(PW||N_c)$ is known to Eve by performing an off-line password guessing attack. That is, Eve makes a guess at the secret password PW^* from dictionary D and checks if $h(PW||N_c) \stackrel{?}{=} h(PW^*||N_c)$. If it holds, Eve has guessed the correct secret password $PW^* = PW$. Compromise of the user's secret password PW will enable Eve to impersonate U or S freely.

For example, suppose that Eve chooses a random nonce N_e and sends an illegal request message Request(username, N_e) to S in step 1 of Tsai's scheme. Then, S will send a challenge message Challenge (realm, $N_s \oplus h(PW||N_e), h(PW||N_s||N_e)$) to Eve. After receiving the challenge message, Eve can send a response message Response(username, realm, h(username, realm, $h(N_s||PW||N_c)$)) to S by using the compromised user's secret password PW. Then, the server S will authenticate Eve by performing the authentication phase. Therefore, Tsai's scheme is obviously insecure against the Denning-Sacco attack.

3.3 Stolen-verifier attacks

In most existing password authentication schemes, the server stores the user's verifier (e.g. plantext passwords or hashed passwords), rather than the user's bare password, in order to reduce the security of the breach once the server is compromised. Therefore, servers are always the targets of attacker, because numerous customers' secrets are stored in their databases. The stolen-verifier attack [15] means that an attacker who steals a password-verifier from the server can use it directly to impersonate a legitimate user in a user authentication execution. Note that the main purpose of an authentication scheme against the stolen verifier attack is to reduce the immediate danger to the authenticate user. In fact, an attacker who has a password-verifier may further mount a guessing attack.

In Tsai's scheme, the password PW of the user, which is stored in the server, can be eavesdropped and then used to masquerade as the original user. Tsai did not explain the stolen-verifier attack, with regard to obtaining the secret data PW, which is stored in a server. This information can allow an illegitimate user to login to the server as a legitimate user. Suppose an attacker has stolen the password PW in the server. Then, he/she can easily impersonate the legal user or the server by performing the authentication phase. Therefore, Tsai's scheme is insecure against stolen-verifier attacks.

3.4 Perfect forward secrecy

Perfect forward secrecy is a very important security requirement in evaluating a strong protocol. A protocol with perfect forward secrecy assures that even if one entity's long-term key is compromised, it will never reveal any session keys used before. For example, the well-known Diffie-Hellman key agreement scheme [8] can provide perfect forward secrecy. Tsai's authentication scheme, however, does not provide it because once the secret password PW of the user U is disclosed, all previous fresh session keys SK will also be opened and hence previous communication messages will be learned.

In Tsai's scheme, suppose an attacker Eve obtains the secret password PW from the compromised user and intercepts transmitted values N_c and $N_s \oplus h(PW||N_c)$, then Eve can compute $h(PW||N_c)$ and extract the server S's random nonce N_s by computing $N_s = N_s \oplus h(PW||N_c) \oplus h(PW||N_c)$. We know that the extracted nonce N_s is the same as the common shared session key between U and S. Therefore, Eve can get the session key $SK = N_s$. Obviously, Tsai's scheme does not provide perfect forward secrecy.

4 Proposed authentication scheme

This section proposes an improved authentication scheme by providing perfect forward secrecy in order to overcome the above mentioned problems with Tsai's authentication scheme.

The proposed scheme can gain benefits from the key block size, speed, and security. The improved scheme consists of three phases; the system setup phase, the registration phase and the authentication phase. Fig.2 illustrates the proposed authentication scheme and it proceeds as follows:

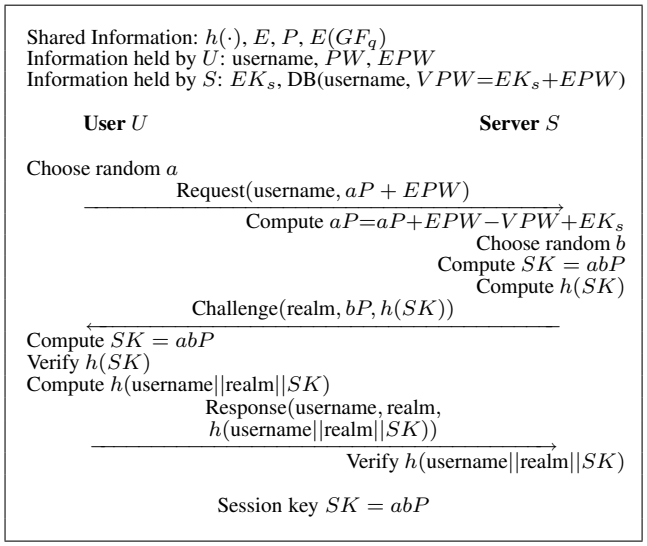

Figure 2. Proposed authentication scheme

4.1 System setup phase

In the system setup phase, U and S agree the following system parameters: U and S choose an elliptic curve E over a finite field $GF(q)$. Let $E(GF_q)$ be an additive group of points on an elliptic curve E over a finite field $GF(q)$. Let P be the generating element(point) of $E(GF_q)$.

4.2 Registration phase

When a user U wants to register and become a new legal user, U and S execute the following steps:

1. $U \rightarrow S$: username, PW
   U submits his/her username and password PW to the remote server S.
2. S computes secret value EPW which is an elliptic curve point in $E(GF_q)$ from the password PW.
3. S computes $VPW = EK_s + EPW$ by using its secret key EK_s and stores the U's username and VPW in the user account database.

4.3 Authentication phase

If a legal user U wants to login into the remote server S, he/she first inputs his/her username and password PW into the client system. Then, the authentication phase proceeds as follows:

1. $U \rightarrow S$: Request(username, $aP + EPW$)
   U generates a random integer a, computes $aP + EPW$, and then sends it with a request message as Request(username, $aP + EPW$) to S.
2. $S \rightarrow U$: Challenge(realm, $bP, h(SK)$)
   Upon receiving the request message, S derives aP by computing $aP + EPW - VPW + EK_s$. Then, S generates a random integer b, and computes a secret session key $SK = abP$ and a message authentication code $h(SK)$. Finally, S sends a challenge message Challenge(realm, $bP, h(SK)$) to U.
3. $U \rightarrow S$: Response(username, realm, h(username||realm ||SK))
   Upon receiving the challenge message, U computes a secret session key $SK = abP$. Then, U computes $h(SK)$ and verifies whether it is equal to the received challenge $h(SK)$. If they are not equal, U rejects the server challenge message. Otherwise, U authenticates S and computes a message authentication code h(username||realm||SK). Finally, U sends a response message Response(username, realm, h(username||realm||SK)) to S.
4. Upon receiving the response message, S computes h(username||realm||SK) and verifies whether it is equal to the received response h(username|| realm||SK). If they are not equal, S rejects the user response message. Otherwise, S authenticates U and accepts the user's login request.

After mutual authentication between U and S, $SK = abP$ is used as a session key.

5 Security analysis

This section provides the security analysis of the proposed authentication scheme. First, we define the security terms [13][14] needed for security analysis of the proposed scheme as follows:

Definition 1 *A weak secret (Password PW) is a value of low entropy $Weak(k)$, which can be guessed in polynomial time.*

Definition 2 *A strong secret (Secret EK_s) is a value of high entropy $Strong(k)$, which can not be guessed in polynomial time.*

Definition 3 *The Elliptic Curve Discrete Logarithm Problem (ECDLP) is as follows: given a public key point $Q = \alpha P$, it is hard to compute secret key α.*

Definition 4 *The Elliptic Curve Diffie-Hellman Problem (ECDHP) is as follows given point elements αP and βP, it is hard to find $\alpha\beta P$.*

Definition 5 *A secure one-way hash function $y = h(x)$ is one where given x to compute y is easy and given y to compute x is hard.*

Here, ten security properties [13][14]: replay attack, password guessing attack, man-in-middle attack, modification attack, Denning-Sacco attack, stolen-verifier attack, mutual authentication, known-key security, session key security, and perfect forward secrecy, must be considered for

the proposed scheme. Under the above definitions, the following theorems are used to analyze nine security properties in the proposed scheme.

Theorem 1 *Proposed scheme can resist the replay attack.*

Proof: Suppose an attacker Eve intercepts Request $(\text{username}, aP + EPW)$ from U in step 1 and replays it to impersonate U. However, Eve cannot compute a correct session key SK and deliver it to S in step 3 unless he/she can correctly guess password PW to obtain aP and guess the right b from bP. When Eve tries to guess a from aP or b from bP, he/she will face the ECDLP. On the other hand, suppose Eve intercepts Challenge$(\text{realm}, bP, h(SK))$ from S in step 2 and replays it to impersonate S. For the same reason, if Eve cannot gain the correct a from $aP + EPW$, U will find out that $h(SK)$ is not equivalent to his/her computed $h(SK)$. Then, U will not send Response$(\text{username}, \text{realm}, h(\text{username}||\text{realm}||SK))$ back to Eve in step 3.

Theorem 2 *Proposed scheme can resist the password guessing attacks.*

Proof: An on-line password guessing attack cannot succeed since S can choose appropriate trail intervals. On the other hand, in an off-line password guessing attack, Eve can try to find out a weak password by repeatedly guessing possible passwords and verifying the correctness of the guesses based on information obtained in an off-line manner. In our scheme, Eve can gain the knowledge of $aP + EPW$, bP, $h(SK)$ and $h(\text{username}||\text{realm}||SK)$ in steps 1, 2, and 3, respectively. To obtain the password PW of U, Eve first guesses password PW^* and then finds $a^*P = aP + EPW - EPW$. By using a^*P and bP, Eve will try to compute the session key $SK = a^*bP$. However, Eve has to break the ECDLP and ECDHP to find the keying material $SK = a^*bP$ from a^*P and bP to verify his/her guess. But, Eve cannot gain the session key without a^* of a^*P and b of bP.

Theorem 3 *Proposed scheme can resist the man-in-middle attack.*

Proof: A mutual password PW between U and S is used to prevent the man-in-middle attack. The illegal attacker Eve cannot pretend to be U or S to authenticate the other since he/she does not own the mutual password PW.

Theorem 4 *Proposed scheme can resist the modification attack.*

Proof: Eve may modify the communication messages $aP + EPW$, bP, $h(SK)$ and $h(\text{username}||\text{realm}||SK)$ being transmitted over an insecure network. However, although Eve forges them, the proposed scheme can detect this modification attack, because it can verify not only the equality of $SK = abP$ computed by each party, but also the correctness of $aP + EPW$ and bP transmitted between two parties through validating $h(SK)$ and $h(\text{username}||\text{realm}||SK)$ in the proposed scheme.

Theorem 5 *Proposed scheme can resist the Denning-Sacco attack.*

Proof: Although an attacker Eve obtains the fresh session key $SK = abP$, Eve cannot obtains the user's secret password PW from $aP + EPW$ because Eve will face the ECDLP by Definition 2 to obtain a from abP.

Theorem 6 *Proposed scheme can resist the stolen-verifier attack.*

Proof: Servers are always the target of attacks. Eve may acquire $VPW = EK_s + EPW$ stored in S. However, without knowing S's secret key EK_s, Eve cannot forge a login request to pass the authentication, as EPW is hidden in $VPW = EK_s + EPW$ using S's secret key EK_s, thus the correctness of the guessed password EPW^* cannot be verified by checking $EPW^* = EPW$.

Theorem 7 *Proposed scheme provides mutual authentication.*

Proof: Mutual authentication means that both the user and server are authenticated to each other within the same protocol, while explicit key authentication is the property obtained when both implicit key authentication and key confirmation hold. As such, the proposed scheme uses the Elliptic Curve Diffie-Hellman key exchange algorithm to provide mutual authentication, then the key is explicitly authenticated by a mutual confirmation fresh session key $SK = abP$.

Theorem 8 *Proposed scheme provides known-key security.*

Proof: Known-key security means that each run of an authentication and key agreement scheme between two entities U and S should produce unique secret keys; such keys are called session keys. Knowing a session key $SK = abP$ and the random values a and b are of no use for computing the other session keys $SK' = a'b'P$, since without knowing a' and b' it is impossible to compute the session key SK'.

Theorem 9 *Proposed scheme provides session key security.*

Proof: Session key security means that at the end of the key exchange, the session key is not known by anyone but U and S. The session key $SK = abP$ is not known by anyone but U and S since the random values a and b are protected by the ECDLP, ECDHP, and the secure one-way hash function. None of this session key $SK = abP$ is known to anybody but U and S.

Theorem 10 *Proposed scheme provides perfect forward secrecy.*

Proof: Perfect forward secrecy means that if long-term private keys of one or more entities are compromised, the secrecy of previous session keys established by honest entities is not affected. If the user's password PW is compromised, it does not allow an attacker Eve to determine the session key SK for past sessions and decrypt them, since Eve is still faced with the ECDHP.

The security properties of previous related schemes, and the proposed scheme are summarized in Table 1.

Table 1. Comparisons of security properties

	Yang et al.'s scheme	Durlanik et al.'s scheme	Tsai's scheme	Proposed scheme
Replay attack	Secure	Secure	Secure	Secure
Password guessing attack	Secure	Secure	Insecure	Secure
Man-in-middle attack	Secure	Secure	Secure	Secure
Modification attack	Secure	Secure	Secure	Secure
Denning-Sacco attack	N/A	Insecure	Insecure	Secure
Stolen-verifier attack	Insecure	Insecure	Insecure	Secure
Mutual authentication	Provided	Provided	Provided	Provided
Known-key security	N/A	Provided	Provided	Provided
Session key security	N/A	Provided	Provided	Provided
Perfect forward secrecy	N/A	Provided	N/A	Provided

6 Performance comparison

The computation costs of the proposed scheme and previous related schemes are summarized in Table 2. The elliptic curve discrete logarithm problem (ECDLP) with an order of 160 bit prime offers approximately the same level of security as the discrete logarithm problem(DLP) with 1024 bit modulus [13].

The proposed scheme requires four ECC multiplications and four hash operations during the protocol. Four ECC computations are needed to prevent a Denning-Sacco attack and to provide perfect forward secrecy. When considering hashing and exclusive-or operations, the proposed scheme requires just four hashing operations. Exclusive-or operations are not required for authentication. Obviously, the proposed scheme is more efficient than previous related authentication schemes for session initiation protocol.

Table 2. Comparisons of computation costs

	Yang et al.'s scheme	Durlanik et al.'s scheme	Tsai's scheme	Proposed scheme
# of exponentiations	4	0	0	0
# of ECC computations	0	4	0	4
# of hash functions	8	8	7	4
# of exclusive-or	4	4	3	0
Security	DLP	ECDLP	Hash	ECDLP

7 Conclusions

We have demonstrated the vulnerabilities of Tsai's nonce-based authentication scheme for session initiation protocol to off-line password guessing attacks, Denning-Sacco attack and stolen-verifier attacks, and also pointed out it does not provide perfect forward secrecy. Then, to resolve such security problems, we presented a new authentication scheme based on the elliptic curve discrete logarithm problem (ECDLP) for session initiation protocol. As a result, the proposed authentication scheme resists those attacks, while also providing more security and efficiency which can be executed faster than other previously proposed related schemes including the Tasi's scheme.

Acknowledgements

We would like to thank the anonymous reviewers for their helpful comments in improving our manuscript. This work is supported by the 2nd Brain Korea 21 Project in 2008.

References

[1] J. Franks, et al., HTTP authentication: basic and digest access authentication, IETF RFC2617, June 1999.

[2] M. Handley, and et al., SIP: session initiation protocol, IETF RFC2543, March 1999.

[3] M. Thomas, SIP Security Requirements, IETF Internet Draft (draftthomas-sip-sec-reg-00.txt), Nov. 2001 (work in progress).

[4] J. Rosenberg, et al., SIP: session initiation protocol, IETF RFC3261, June 2002.

[5] J. Arkko, et al., Security mechanism agreement for SIP sessions, IETF Internet Draft (draft-ietf-sipsec-agree-04.txt), June 2002.

[6] L. Veltri, S. Salsano, and D. Papalilo, SIP security issues: the SIP authentication procedure and its processing load, IEEE Network, vol. 16, no. 6, pp. 38-44, 2002.

[7] C. C. Yang, R. C. Wang, and W. T. Liu, Secure authentication scheme for session initiation protocol, Computers and Security, vol. 24, pp. 381-386, 2005.

[8] W. Diffie, and M. Hellman, New directions in cryptology, IEEE Transaction on Information Theory, vol. 22, no. 6, 1976.

[9] A. Durlanik, and I. Sogukpinar, SIP authentication scheme using ECDH, World Enformatika socity Transaction on Engineering computing and technology , vol. 8, pp. 350-353, 2005.

[10] N. Koblitz, Elliptic curve cryptosystems, Mathematics of Computation, vol. 48, pp. 203-209, 1987.

[11] NIST, Recommended Elliptic Curves for Federal Government Use, July 1999.

[12] J. L. Tsai, Efficient nonce-based authentication scheme for session initiation protocol, International Journal of Network Security, vol. 8, no. 3, pp. 312-316, May 2009.

[13] A. J. Menezes, P. C. Oorschot, and S. A. Vanstone, Handbook of applied cryptograph, CRC Press. New York, 1997.

[14] D. Denning, and G. Sacco, Timestamps in key distribution systems, Communications of the ACM, vol. 24, pp. 533-536, 1981.

[15] C. L. Lin, and T. Hwang, A password authentication scheme with secure password updating, Computers and Security. vol. 22, No. 1, pp. 68-72, 2003.