A New Provably Secure Authentication and Key Agreement Protocol for SIP Using ECC*

Liufei Wu[1，2], Yuqing Zhang[1], Fengjiao Wang[1]

[1](National Computer Network Intrusion Protection Center, GUCAS, Beijing, China 100049)

[2](Communication Engineering Institute, Xidian University, Xi'an, China 710071)

Abstract

SIP is playing a key role in the IP based services and has been chosen as the protocol for multimedia application in 3G mobile networks by the Third-Generation Partnership Project. The authentication mechanism proposed in SIP specification is HTTP digest based authentication, which allows malicious parties to impersonate other parties or to charge calls to other parties, furthermore, other security problems, such as off-line password guessing attacks and server spoofing, are also needed to be solved. This paper proposes a new authenticated key exchange protocol NAKE, which can solve the existed problems in the original proposal. The NAKE protocol is provably secure in CK security model, thus it inherits the corresponding security attributes in CK security model.

Key words: SIP; NAKE protocol; CK security model; Provable security

1 Introduction

The Internet Engineering Task Force (IETF) proposed the Session Initiation Protocol (SIP) as the IP-based telephony protocol [1]. SIP is a text based protocol with similar formatting to HTTP capable of operating on TCP or UDP and handles all the signaling requirements of a Voice over IP (VoIP) session, which is analogous to the SS7 protocol in traditional telephony [2]. It is an application-layer signaling protocol for creating, modifying, and terminating multimedia sessions with one or more participants [1]. Currently, the security of SIP is becoming more and more important. SIP specification does not include any specific security mechanisms. SIP authentication is inherited from HTTP Digest authentication, which is a challenge-response based authentication protocol.

1.1 History and Related Work

Security and privacy requirements in a VoIP environment are expected to be equivalent to those in Public switched telephone network (PSTN), even though the provision of secure Internet services is much more complicated. Salsano et al pointed out that the HTTP digest authentication in SIP suffers from two major weaknesses when it is applied in SIP [3]. One is the lack of securing all headers and parameters in SIP which would possibly need protection. The other is the requirement of pre-existing user configuration on servers. Furthermore, a methodology for the evaluation of the processing cost of SIP authentication procedure is also given in their work, which is of great meaning. HTTP Digest authentication scheme in SIP can offer one-way message

* This work is supported by National Natural Science Foundation of China (grant Nos. 60573048, 60373040).

[1]Corresponding author: Tel: +86-010-68860988, Fax: +86-010-68860988.

E-mail address: wlf0701@hotmail.com

authentication and replay protection but not the support message integrity and confidentiality. According to RFC 3261 [1], it is very possible for a malicious user to place spam calls and send a manipulated message to cause a Dos. Moreover, this method is vulnerable to well known plaintext, replay attack, off-line password guessing attack, man-in-the-middle attacks and server spoofing [4, 5].

Since the current authentication mechanism is not providing security at an acceptable level, several new schemes are proposed to improve it. An identity-based authentication and key agreement protocol without formal analysis was presented in [6], and the off-line password guessing attack and server spoofing attack to original authentication mechanism have been found. A SIP authentication scheme by using a public key exchange mechanism using Elliptic Curve Cryptography (ECC) was proposed in [7]. It has significant advantages like smaller key sizes, faster computations on behalf of other Public Key Cryptography systems that obtain data transmission more secure and efficient. However, it is vulnerable to man-in-the-middle attack, and does not reflect the good characteristics of Elliptic Curve Diffie-Hellman (ECDH) protocol.

1.2 Our Solution

This paper proposes the use of ECC cryptography as a solution to the authentication and key agreement problems that exist in SIP. This new SIP authentication mechanism and key agreement protocol provides mutual authentication and provable security in Canetti-Krawczyk (CK) security model. This solution fits neatly in the SIP protocols as described in RFC 3261 [1].

1.3 Outline of Paper

Section 2 presents some background information about SIP authentication scheme, RFC3310. In section 3, the new SIP key agreement protocol is shown. The security proof is given in section 4. This is followed by a discussion of the proposed solution and observed limitations in Section 5. Finally, Section 6 concludes the paper.

2 SIP Authentication Procedure

SIP authentication [2] security is based on the challenge-response mechanism, in which a nonce value is used in challenging the target. Before the scheme starts, the client pre-shares a password with the server. SIP applies the digest mechanism for authenticating users to users or users to proxies, not proxies to proxies. The security between proxies relies on other mechanisms, for example TLS or IPsec. Figure 1 is an example flow of authentication mechanism in SIP.

Step 1. client → server: REQUEST

The client sends a REQUEST to the server.

Step 2. server → client: CHALLENGLE(nonce, realm)

The server sends an response message containing a nonce value and a realm to the client. The response is actually an error message requesting authentication. The realm in the message is the digest algorithm used in this challenge.

Step 3. client → server: RESPONSE(nonce, realm, username, response)

The client computes the response, which is computed with nonce value received in challenge, a username and a secret password. Then the client sends back the original request message with the computed response value, username, nonce value and realm.

Step 4. According to the username, the server extracts the client’s password. Then the server verifies whether the nonce is correct. If it is correct, the server computes F (nonce, username, password, realm) and compare it with the response. If they match, the server authenticates the identity of the client.

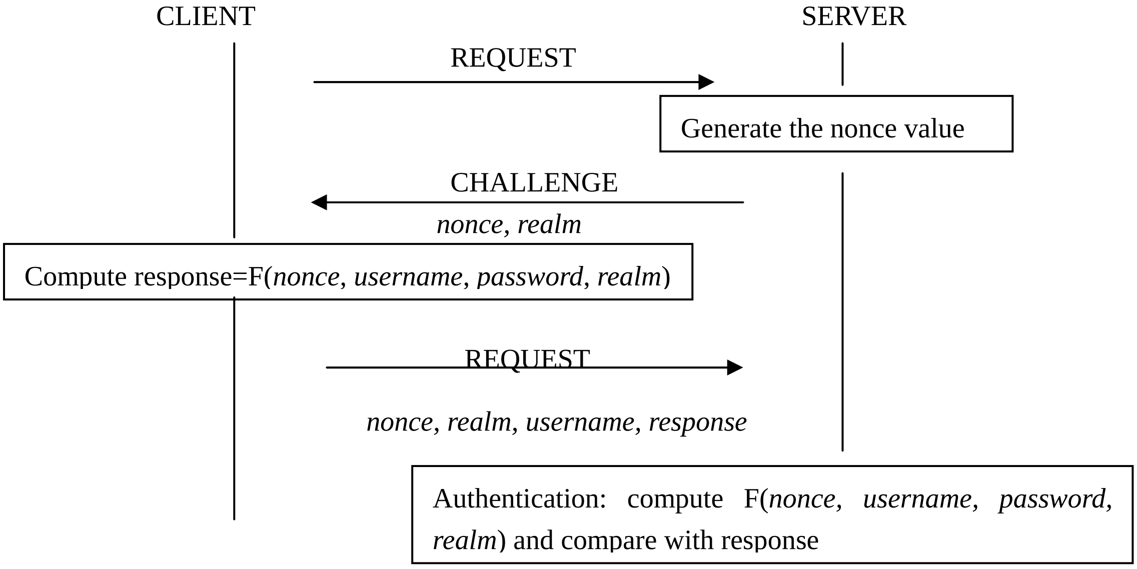
CLIENT
SERVER
REQUEST
Generate the nonce value
CHALLENGE
nonce, realm
Compute response=F(*nonce*, *username*, *password*, *realm*)
REQUEST
nonce, realm, username, response
Authentication: compute F(*nonce, username, password, realm*) and compare with response

Figure 1 Digest authentication procedure in SIP

3 A New Authentication and Key Exchange Protocol for SIP

This new authentication and key exchange protocol for SIP uses the provably secure implementation of ECDH protocol and provides authentication of both parties at the end of the handshake, while ensuring that the operation remains consistent with the requirements of RFC3261. Figure 2 is a message flow describing a Digest AKE process of authenticating a SIP REGISTER request.

Client $\rightarrow$ Server: REGISTER
Server $\rightarrow$ Client: 401: Unauthorized $\{server, realm, oopaque, fnonce\}$
Client $\rightarrow$ Server: Authorization $\left\{client, realm, opaque, fnonce', f\left(k, realm, client, opaque, nonce', nonce, server\right)\right\}$
Server $\rightarrow$ Client: 200 OK $\left\{client, realm, opaque, f\left(k, realm, server, opaque, nonce, nonce', client\right)\right\}$

Figure 2 NAKE protocol

3.1 Initial Information

The correct and secure operation of this protocol depends upon a number of realistic assumptions. It is assumed that all parties have globally agreed upon a nonsingular high elliptic curve, $E\left(F_q\right)$, where $P \in E\left(F_q\right)$, and be of order n , such as would be defined by an appropriate standards body. P_S , P_C denotes the identities of client and server respectively, opaque string is used as a session identifier, denotes by s . f is a hash function. Furthermore it is assumed that all private keys remain private and secure, and that nonce will not be reused.

3.2 Description

The proposed authentication and key agreement operation is similar to the HTTP Digest using AKA Mechanism, and as stated can operate without changing the semantics of RFC3310. The handshake is described here:

1. A shared secret k and P are established beforehand between the ISIM and the Authentication Center (AuC).

k is stored in the ISIM, which resides on a smart card like, tamper resistant device.

2. Client makes request of a SIP service requiring authentication (REGISTER).
3. Server chooses ephemeral private key x , $x \in_R [1, n-1]$ and calculates nonce $\alpha = x \bullet P$, fnonce $\alpha \oplus f(k)$. Then, it prepares 401 Unauthorized Authentication Required as appropriate. This response is a challenge consisting of: a realm string, fnonce, opaque string, and identity of the Server. An opaque string is used as a session identifier，noted s .
4. Client verifies the server response using the shared secret k . Client then chooses an ephemeral private key y , and calculates nonce $\beta = y \bullet P$, fnonce $\beta \oplus f(k)$ and produces an authentication response $f(k, P_C, s, \beta, \alpha, P_S)$. Then, client derives session key $sk = y \bullet \alpha$.
5. Client sends the response along with their username, realm, nonce and opaque string in clear text to the entity requesting authentication.
6. Server verifies the response with k respectively, and prepares an authentication response $f(k, P_s, s, \alpha, \beta, P_C)$. Thus, server derives session key $sk = x \bullet \beta$.
7. Server responds with appropriate error message or grants access. If grants access, the responds consist of the realm, opaque string and the hash value based on k .

3.3 CK Security Model

CK security model [8] presents definition of SK-security, allows for modular design and analysis of key exchange protocol, which simplifies the difficulty of design and analysis of security protocol. The security definition is based on the concept of indistinguishability

The attacker model follows the unauthenticated-links model (UM) that the attacker is a (probabilistic) polynomial-time machine with full control of the communication lines between parties. In addition, the attacker can have access to secret information via session exposure attacks of three types: session-state reveal, session-key queries, and party corruption. The first type of attack is directed at a single session which is incomplete and the result is that the attacker learns the session state of that particular session. A session-key query can be performed against an individual session after completion and the result is that the attacker learns the corresponding session-key. Finally, party corruption means that the attacker learns all information in the memory of that party; in addition, from the moment a party is corrupted all its actions are totally controlled by the attacker.

Sessions can be expired in the model of CK. From the time a session is expired the attacker is not allowed to perform a session-key query or a state-reveal attack against the session, but is allowed to corrupt the party that holds the session. Protocols that ensure that expired sessions are protected even in case of party corruption are said to enjoy “perfect forward secrecy”.

For defining the security of a KE protocol, CK follows the indistinguishability style of definitions that the “success” of an attacker is measured via its ability to distinguish the real values of session keys from independent random values. When the attacker chooses the test session it is provided with a value υ which is chosen as follows: a random bit b is tossed, if $b = 0$ then υ is the real value of the output session-key, otherwise υ is a random value chosen under the same distribution of session-keys produced by the protocol, but independent of the value of the real session key. After receiving υ , the attacker may proceed with the regular actions against the protocol; at the end of its run the attacker outputs a bit b' . The attacker succeeds in its attack if (1) the test session is not exposed, and (2) the probability that $b = b'$ is significantly larger than $1/2$. Note that the attacker is allowed to corrupt a party to the test session once the test expires at that party (this captures perfect forward secrecy).

An adversarial model called authenticated-links model (AM) is defined in a way that is identical to the UM with one fundamental difference: the attacker is restricted to only deliver messages truly generated by the parties without any change or addition to them. Then the notion of “emulation” is introduced in order to capture the equivalence of functionality between protocols in different adversarial models, in particular between the UM and AM.

The resultant security notion for KE protocols is called SK-security and is stated as follows:

Definition 1. (SK-security) An attacker with the above capabilities is called an SK-attacker. A key-exchange protocol π is called SK-secure if for all SK-attacker $\mathcal{A}$ running against π it holds:

1. If two uncorrupted parties complete matching sessions in a run of protocol π under attacker $\mathcal{A}$ then, except for a negligible probability, the session key output in these sessions is the same.
2. $\mathcal{A}$ succeeds in its test-session distinguishing attack with probability not more than $1/2$ plus a negligible fraction.

3.4 Security Proof of NAKE Protocol

We first demonstrate that under the Elliptic Curve Decisional Diffie-Hellman (ECDDH) assumption the classic two-move Elliptic Curve Diffie-Hellman key-exchange protocol designed to work against an eavesdropper only is SK-secure in the AM. We denote this protocol by ECDH and describe it in Figure 2. Using Theorem 1 we can apply an appropriate authenticator to this protocol to obtain a secure Elliptic Curve Diffie-Hellman exchange against realistic UM attackers.

common information：prime $P \in E\left(F_q\right)$ of order n 。

goal：server and client share a session key: $sk = x{\bullet}\beta = y{\bullet}\alpha$

1. server → client： $s, P_S, \alpha \oplus f\left(k\right)$ ， $\alpha = x{\bullet}P$
2. client → server： $s, P_C, \beta \oplus f\left(k\right)$ ， $\beta = y{\bullet}P$

Figure 3 ECDH protocol

The Elliptic Curve Decisional Diffie-Hellman (ECDDH) assumption is as follows:

ECDDH Assumption Let E be a nonsingular high Elliptic Curve on finite field F_q , Let $P \in E\left(F_q\right)$ be of order n , $x, y, z \in_R [1, n-1]$. Then the probability $\mathcal{D}$ distributes quintuples $Q_0 = \langle P, x{\bullet}P, y{\bullet}P, x{\bullet}y{\bullet}P \rangle$ and $Q_1 = \langle P, x{\bullet}P, y{\bullet}P, z{\bullet}P \rangle$ is computationally indistinguishable.

Theorem 2 If ECDDH assumption holds, protocol ECDH is SK-secure in the AM.

Proof: To see that the first requirement of Definition 1 is satisfied, note that if both parties are uncorrupted during the exchange of the key and both complete the protocol then they both establish the same key (which is sk). Note that the session identifier s uniquely binds the values of $\alpha \oplus f\left(k\right)$ and $\beta \oplus f\left(k\right)$ to these particular matching sessions and differentiates them from other exponentials that the parties may exchange in other sessions.

We show that the second requirement of Definition 1 is also satisfied by protocol ECDH. Assume to the contrary that there is a KE-adversary $\mathcal{A}$ in the AM against protocol ECDH that has a non-negligible advantage ε in guessing correctly whether the response to a test-query is real or random. Out of this attacker $\mathcal{A}$, we construct an algorithm $\mathcal{D}$ that distinguishes between the distributions Q_0 and Q_1 with non-negligible probability, thus reaching a contradiction with Assumption 1. Algorithm $\mathcal{D}$ uses adversary $\mathcal{A}$ as a subroutine and is described in Figure 4.

Proceed as follows, on input $\left\langle q, P, \alpha^*, \beta^*, \gamma^* \right\rangle$:

1. Choose $r \leftarrow_R \{1 \ldots l\}$.

2. .Invoke $\mathcal{A}$ on a simulated interaction in the AM with parties running ECDH. Hand $\mathcal{A}$ the values, q , P as the public parameters for the protocol execution.
3. Whenever $\mathcal{A}$ activates a party to establish a new session (except for the r-th session) or to receive a message, follow the instructions of ECDH on behalf of that party. When a session is expired at a player erase the corresponding session key from that player's memory. When a party is corrupted or a session (other than the r-th session) is exposed, hand $\mathcal{A}$ all the information corresponding to that party or session as in a real interaction.
4. When the r-th session, say $\left(P_S, P_C, s\right)$, is invoked with P_S to exchange a key with P_C , let P_S send the message $\left(P_S, s, \alpha^*\right)$ to P_C .
5. When P_C is invoked to receive $\left(P_S, s, \alpha^*\right)$, let P_C send the message $\left(P_C, s, \beta^*\right)$ to P_S .
6. If session $\left(P_S, P_C, s\right)$ is chosen by $\mathcal{A}$ as the test-session, then provide $\mathcal{A}$ with γ^* as the answer to this query.
7. If the r-th session $\left(P_S, P_C, s\right)$ is ever exposed, or if a session different than the r-th session is chosen as the test-session, or if $\mathcal{A}$ halts without choosing a test-session then $\mathcal{D}$ output $b' \leftarrow_R \{0,1\}$ and halts.
8. If $\mathcal{A}$ halts and outputs a bit b' , then $\mathcal{D}$ halts and output b' too.

Figure 4 Distinguisher $\mathcal{D}$

First note that the run of $\mathcal{A}$ by $\mathcal{D}$ is identical to a normal run of $\mathcal{A}$ against protocol ECDH.

Consider the case in which the test session coincides with the r-th session, and then the response to the test-query by $\mathcal{A}$ is γ^* . In addition, input to $\mathcal{D}$ was chosen with probability that $1/2$ from Q_0 and Q_1 , and the advantage that $\mathcal{A}$ guesses correctly whether the test value was "real" or "random" is ε . Thus the distinguisher $\mathcal{D}$ guesses correctly the input distribution Q_0 or Q_1 with the same probability $1/2+\varepsilon$ as $\mathcal{A}$ did.

Now consider the case in which the r-th session is not chosen as a test-session. In this case $\mathcal{D}$ always ends outputting a random bit, and thus its probability to guess correctly the input distribution is $1/2$.

Since the first case happens with probability $1/l$ while the other case happens with probability $1-1/l$ we get that the overall probability of $\mathcal{D}$ succeeds in distinguishing Q_0 from Q_1 with non-negligible advantage.

1. server $\rightarrow$ client： m
2. client $\rightarrow$ server： m, N_C
3. server $\rightarrow$ client： $m, f\left(k, m, N_C \oplus f\left(k\right), P_C\right)$

Figure 5 Pre-shared key based MT-authenticator

Applying the signature-based authenticator in figure 5 to each of the flows in ECDH protocol and joining (piggy-baking) the common flows，then we can get protocol NAKE in UM. Follows from Theorems 1 and 2, NAKE is a SK-secure protocol under UM. □

4 Discussion

The new SIP authentication mechanism and key agreement protocol proposed here meets the goal and requirements stated above. The cryptographic primitive used to provide the assurances are provably secure in CK security model. Non-repudiation, protection against replay and session hijacking attacks, and mutual authentication are by-products from the use of ECC cryptography and hash value. We analyze the security of our scheme as follows.

Replay attack

Replay attacks cannot work in this scheme can be provided by the freshness of session id s .

Off-line password guessing attack

The attacker guess a password k and computes $f\left(k\right)$ .Then, the attacker computes

$f\left(k,P_C,s,\left(f(k)\oplus\left(f(k)\oplus\beta\right)\right),\left(f(k)\oplus\left(f(k)\oplus\alpha\right)\right),P_S\right)$ and $f\left(k,P_s,s,\left(f(k)\oplus\left(f(k)\oplus\alpha\right)\right),\left(f(k)\oplus\left(f(k)\oplus\beta\right)\right),P_C\right)$. Obviously, the attacker cannot compute the value k to match the RESPONSE, because it faces the difficulty of discrete logarithms. Therefore, the protocol is immune to the off-line password guessing attack.

Server spoofing

The server computes shared key $sk = x\bullet\beta$ and sends $f\left(k,P_s,s,\alpha,\beta,P_C\right)$ to the client. The client can verify the identity of the server by computing $f\left(k,P_s,s,\alpha,\beta,P_C\right)$. Thus, the attacker cannot impersonate the server to deceive the client. Meanwhile, the client derives shared key by computing $sk = y\bullet\alpha$ and sends $f\left(k,P_C,s,\beta,\alpha,P_S\right)$ to the server. Then, the server can verify the identity of client.

Mutual authentication

Both parties produce a hash value based on pre-shared key for mutual authentication, and meet the security objectives of mutual authentication.

Mutual key agreement and control

Protocol AKE based on Diffie-Hellman key exchange, freshness of session key to ensure appropriate selection of random numbers. Two sides enjoy a separate key based on hash value produced by pre-shared key. Security parameters α and β were each randomly selected by the server and client. Thus, server and client are beyond the control of key generation.

Mutual key confirm

After the end of the protocol by the hash value $f\left(k,P_s,s,\alpha,\beta,P_C\right)$ and $f\left(k,P_C,s,\beta,\alpha,P_S\right)$ of server and client respectively, moreover the two sides can ensure that they have a specific key.

Perfect forward secrecy

Session key is established by Diffie-Hellman key exchange, thus, AKE protocol have the attractive property of PFS.

Furthermore, its total execution times and memory requirements of proposed scheme have been improved in comparison with non-elliptic approaches by adopting elliptic-based key exchange mechanism. It is more efficient and preferable in the applications which require low memory and rapid transactions. However, this new scheme has its limitations. It requires a pre-arranged trusted environment for password distribution as Digest authentication does. But it is fortunate that this new protocol could easily be extended to PKI circumstances, which would have the same security attributes.

5 Conclusion

This paper proposes a new authentication scheme for SIP, which overcomes the inherent weaknesses of AKA scheme, achieves the authentication and a shared secret at the same time and provides provable security in CK security model. This solution fits neatly in the SIP protocols as described in RFC3261 .The new protocol has security attributes required by SIP standard and requires only minimal changes to the standard. The scheme is designed to provide data confidentiality, data integrity, authentication, access control, and perfect forward secrecy, and it is secure against known-key attacks. Moreover, NAKE is based on ECC, so it is more efficient and preferable in the applications which require low memory and rapid transactions.

Acknowledgments

The authors would like to thank the anonymous reviewers for their constructive comments. This research was partially supported by National Natural Science Foundation of China (grant Nos. 60573048, 60373040).

References

[1] Rosenberg, J, Schulzrinne, H, Camarillo, G, et al, SIP: Session Initiation Protocol, RFC 3261, June 2002.

[2] Niemi, A, Nokia, Arkko, et al, Hypertext Transfer Protocol (HTTP) Digest Authentication Using Authentication and Key Agreement (AKA). RFC 3310, September 2002.

[3] Salsano Stefano, Veltri Luca and Papalilo Donald, SIP security issues: the SIP authentication procedure and its processing load, *IEEE Network*, Volume 16 (Issue 6, Nov-Dec 2002) 38-44.

[4] Yang C C, Wang R C, Liu W T. Secure authentication scheme for session initiation protocol, Computer &Security, 2005, vol.24: 381-386.

[5] Chia-Chen Chang, Yung-Feng Lu, Ai-Chun Pang etc.Design and Implementation of SIP Security. C.Kim (Ed.): ICOIN 2005,LNC 3391, (Springer, Berlin, 2005) 669-678.

[6] Jared Ring, Kim-Kwang Raymond Choo, Ernest Foo, et a, A New Authentication Mechanism and Key Agreement Protocol for SIP Using Identity-based Cryptography, AusCERT2006 R&D Stream. Gold Coast, Australia, 2006.

[7] Aytunc Dulanik, Ibrahim Sogukpinar, SIP Authentication Scheme using ECDH, In: ENFORMATIKA, 2005, Vol (V8 2005 ISSN 1305-5313) 350-353.

[8] Canetti R, Krawczyk H, Analysis of Key-Exchange Protocols and Their Use for Building Secure Channels, In: Pfitzmann ed.Proceedings of Eurocrpt'01, Lecture Notes in Computer Science 2045. (Springer, Berlin, 2001) 453-474.