A New Secure Authentication and Key Exchange Protocol for Session Initiation Protocol Using Smart Card

Mourade Azrour, Yousef Farhaoui, Mohammed Ouanan

(Corresponding author: Mourade Azrour)

Department of Computer Science, M2I Laboratory, ASIA Team, Moulay Ismail University
BP 509, Boutalamine, 52000 Errachidia, Morocco
(Email: azrour.mourade@gmail.com)

(Received July. 26, 2016; revised and accepted Nov. 15 & Dec. 25, 2016)

Abstract

In today′s communications over Internet Protocol (IP), Session Initiation Protocol (SIP) is using establish, modify and terminate the sessions multimedia among participants. Authentication is the most security service required for SIP. Authentication HTTP Digest is the original authentication protocol proposed for SIP. However, this protocol is demonstrated insecure against different attacks. To improve the authentication, a different authentication protocols have being proposed. Very recently, Jiang et al. demonstrate that Zhang et al.'s scheme cannot resist to impersonation attack. Then, Jiang et al. proposed their protocol. However, in this paper we show that Jiang et al.'s protocol suffer from server spoofing attack. In order to overcome this problem we propose an improved SIP authentication protocol. The security analysis shows that the proposed protocol is more secure and can deal with several attacks.

Keywords: Authentication Protocol; Elliptic Curve Cryptography; Session Initiation Protocol; Smart Card

1 Introduction

Telephony over IP (ToIP) is a service that allows transferring voice communications flow on IP (Internet Protocol). This is the application that will require the IP infrastructure as the standard for all types of information or media. IP telephony is based on open standards. To establish ToIP communication two types of protocols are required which are signaling protocol and transport protocol. In recently decade, Session Initiation Protocol (SIP) is the most signaling protocol used for establishing, altering and terminating session multimedia between different users.

Authentication is the most security service required by SIP. The original SIP authentication protocol is HTTP Digest Authentication. This protocol was found vulnerable to deferent attacks. In order to reinforce SIP authentication, a large community has been participated by proposing the different protocols based on various mechanisms. Generally, authentication protocol can be categorized as Password Authentication protocol [17, 21, 22], ID-based protocol [7] and Elliptic Curve Cryptography based protocol [2].

In 2005, Yang et al. [25] demonstrated that the original SIP authentication protocol is vulnerable to Off-line password guessing attack and stolen verifier attack. So, they based on Diffie-Hellman Key Exchange [3] to propose their protocol which is secure against Off-line password guessing attack, server spoofing attack and replay attack. However, the protocol of Yang et al. requires maintenance and configuration of the passwords table. In addition, it based on the discrete logarithm problem which requires an important computation time cost. Therefore, it is not suitable for applications with low memory and limited computing capability. In 2006, Huang et al. [9] proposed a new protocol based on one-way hash functions. After comparing the computational complexity of their protocol with the Yang et al.'s protocol they concluded that their protocol is the fastest. Moreover, Jo et al. [12] demonstrated that the protocol of Yang et al. and the protocol of Huang et al. are both vulnerable to Off-line password guessing attack.

To overcome this weakness, Durlanik and Sogukpinar [4] based on the Yang et al.'s protocol to propose another SIP authentication protocol using the Elliptic Curve Cryptography Diffie-Hellman (ECCDH) [13]. They demonstrated that their protocol reduces the computation time cost. Because it uses a small size key but offer the same security offered by the Diffie-Hellman large key size. However, Yoon et al. [27, 29] introduced that the protocol of Durlanik and Sogukpinar cannot resist the stolen verifier attack and Denning-Sacco attack.In 2008, Wu et al. [23] proposed a new SIP authentication and key exchange protocol based

on elliptic curve cryptography (ECC). Wu et al. prove that their protocol is secure against man-in-the-middle attack, replay attack, Off-line password guessing attack and server spoofing attack. Unfortunately, this protocol is vulnerable to Off-line password guessing attack, Denning-Sacco attack and stolen verifier attack [26]. In the same year, Tsai [19] proposed an authentication protocol for SIP based on random nonce. The protocol uses one-way hash functions, and a bit-wise exclusive-or(XOR) operation to encrypt and decrypt messages. As result, the calculation time cost is reduced when it compared with the existing protocols. For this, it is desirable for applications with low computing capability. However, Yoon et al. [28], then Arshad and Ikram [1] found that Tsai′s protocol is vulnerable to Off-line password guessing attack, server spoofing attack and stolen verifier attack. One year later, Yoon and Yoo [28] proposed a new secure SIP authentication protocol. They demonstrated that their protocol is secure against the man-in-the-middle attack, Off-line password guessing attack, replay attack, modification attack, Denning-Sacco attack and stolen verifier attack. In addition, it provides mutual authentication, known key secrecy, session key secrecy and perfect forward secrecy. However, Liu and Koenig [14] demonstrated that this protocol is vulnerable to Off-line password guessing attack and partition attack. In 2011, Arshad and Ikram [1] demonstrated that Tsai et al.'s protocol is vulnerable to Off-line password guessing attack and stolen verifier attack, and it does not provide key known secrecy and perfect forward secrecy. As result, Arshad and Ikram presented an authentication protocol for SIP based on ECC. In 2012, Xie [24] showed that the protocol of Yoon and Yoo is insecure against stolen verifier attack and Off-line password guessing attack. Based on these attacks Xie proposes a new SIP authentication protocol. Then, he demonstrated that his protocol is more secure, and it is faster when it compared with existing protocols. However, Xie′s protocol is shown vulnerable to Off-line password guessing attack. In the same year, Tang et al. [18] noted that the protocol introduced by Arshad and Ikram is not secure against Off-line password guessing attack. In order to deal with this problem, they suggested another secure and efficient SIP authentication protocol based on Elliptic Curve Discrete Logarithm Problem (ECDLP).

In 2013, Zhang et al. [30] introduced for the first time smart-card-based protocol and key exchange for SIP. Then, they demonstrated that their protocol is secured against different attacks. However, Tu et al. [20], Irshad et al. [10], Zhang et al. [31], and Jiang et al. [11] demonstrated that Zhang et al.'s scheme is insecure against impersonation attacks. To solve the problem Jiang et al. [11] proposed a new SIP authentication protocol. Then, they proved that their scheme resist to various attacks. However, in this paper we demonstrate that Jiang et al.'s protocol is vulnerable to server spoofing attack. In order to overcome this weakness, we propose a secure and efficient SIP authentication protocol using smart card and based on elliptic curve cryptography.

The remainder of this paper is organized as follows. Section 2 delivers general information on the architecture and the original SIP authentication protocol. In Section 3, we review briefly Jiang et al.'s scheme. A cryptanalysis of Jiang et al.'s scheme is given in Section 4. In Section 5, we present our secure and efficient SIP authentication protocol. The security analysis and performance comparison are presented in Sections 6 and 7, respectively. Finally, section 8 concludes the paper.

2 Preliminaries

Session Initialization Protocol was initiated by the Multiparty Multimedia Session Control Group (MUSICG) in RFC 2543 [5]; then it was taken over and maintained by the SIP Group of the Internet Engineering Task Force (IETF). The first works are started from 1995, which resulted in a first version of SIP with the publication of RFC 2543 [6] in 1999; then a second version of SIP was published in 2002 to correct certain defects of the previous version.

SIP is a text-based protocol built on the basis of protocols such as HTTP or SMTP. The exchanges are in the form of dialogues (peer-to-peer relationships between agents) that include transactions (request/response). It is a widely used protocol, mainly for telephony applications on IP.

2.1 SIP Architecture

The architecture of SIP consists of a proxy server, redirect server, register server, location server, and User agents. The role of each component is described as follows.

User Agent Client (UAC): generates SIP requests before they were sent;

User Agent Server (UAS): generates answers to SIP requests (accepting, refusing, or redirecting);

User Agent (UA): it can be a SoftPhone (software) or HardPhone (IP phone). It is able to generate, send and receive SIP requests. It can act at the same time as a UAC and UAS;

Registrar Server: handles the registration of SIP terminals. This is a server that accepts SIP REGISTER requests;

Proxy Server: is a server which is connected to fixed or mobile terminals (UA). It plays the role of a server and client;

Redirect Server: is a server that accepts SIP requests, translates the SIP address of a destination network IP address and returns them to the client;

Location Server: The responsibility of the location server is to maintain information on the current location of the user agent. It provides the proxy server, redirect server, and register server, it allows for them to look up or register the location of the user agent.

2.2 HTTP Digest Authentication Protocol

The authentication of SIP is the most security service recommended by the IETF in RFC 2617 [16]. If a user wants to get access into the SIP services, he/she must be authenticated by server. In addition to needing to know if a user's identity is legitimate or not. The user also needs to know if the server with which it communicates is the legal server or not.

HTTP Digest Authentication for SIP is based on the mechanism challenge/response. Before the protocol execution, the client and the server share the password, the latter is used to verify the client′s identity. The messages exchanged between the server and the clients during authentication procedure are illustrated in Figure 1. and they are described as follows:

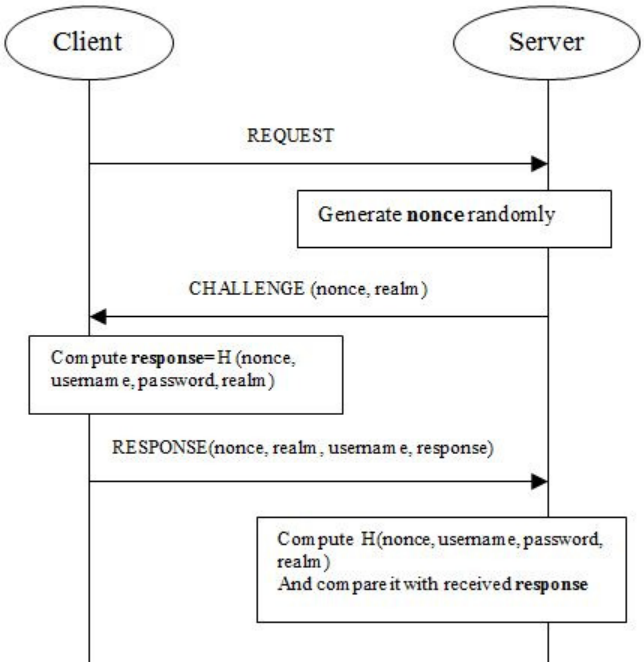

Figure 1: HTTP digest authentication

Step 1. Client → Server: REQUEST

The client sends a REQUEST to the server;

Step 2. Server → Client: CHALLENGE (nonce, realm)

After receiving REQUEST; the server generates CHALLENGE that includes a nonce and the client′s realm. Note that realm is used to verify username and password. Then the server sends back CHALLENGE to the client;

Step 3. Client → Server: RESPONSE (nonce, realm, username, response)

After receiving CHALLENGE from the server, the client computes the response by using received nonce, username, secret password, and realm. response=F(nonce, username, password, realm). Note that $F(\cdot)$ is a one-way hash function. Next, the client sends back the original REQUEST with the computed response, username, nonce and realm;

Step 4. According to username the server extracts the client's password. Then, the server verifies wither nonce is correct or not. If it is correct, the server computes F(nonce, username, password, realm) and uses it to compare it with the response. If they match, the server authenticates the identity of the client.

2.3 Elliptic Curve Cryptography

Elliptic curve cryptography (ECC) was introduced by Neal Koblitz in 1985 [25]. ECC proposed as an alternative to established public-key systems such as DSA and RSA. ECC have lately received a lot attention in information security. The main reason for the attractiveness of ECC is the fact that there is no sub-exponential algorithm known to solve the discrete logarithm problem on a properly chosen elliptic curve. This means that ECC uses the keys of small size but offer the same levels of security offered by the Diffie-Hellman key large size. Some benefits of having smaller key size include faster computations, and reductions in processing power, storage space and bandwidth. This makes ECC ideal for constrained environments such as cellular phones and smart cards [15].

The elliptic curve is a cubic equation of the form in Equation (1):

$$E : y^2 + axy + by = x^3 + cx^2 + dx + e \tag{1}$$

where a, b, c and e are real numbers.

In cryptosystem, the elliptic curve equation is defined as the form in Equation (2) over a prim finite field $(F)_p$, where $(a, b) \in F_p$ and $4a^3 + 27b^2 \neq 0(mod\ p)$. Given an integer $k \in (F_p)^*$ and a point $P \in E_p(a, b)$, the scalar multiplication kP over $E_p(a, b)$ can be computed as in Equation (3).

$$E_p(a, b) : y^2 = x^3 + ax + b(mod\ p) \tag{2}$$

$$kP = (P + P + ... + P)_{(k\ times)} \tag{3}$$

Definition 1. *Given two points P and Q over $(E)_p(a, b)$, the elliptic curve discrete logarithm problem (ECDLP) is to find an integer $k \in (F_p)^*$ such as $Q = kP$.*

Definition 2. *Given three points P,sP and kP over $E_p(a, b)$for s,$k \in (F_p)^*$, the computational Diffie-Hellman problem (DHP) is to find the point skP over $E_p(a, b)$.*

Definition 3. *Given two points P and $Q = sP+kP$ over $E_p(a, b)$ for $s, k \in (F_p)^*$, the elliptic curve factorization*

problem (ECFP) is to find two points sP and kP over $E_p(a,b)$.

3 Review of Zhang et al.'s Scheme

In this section we briefly review the Jiang et al.'s [30] authentication scheme for SIP. The Jiang et al.'s scheme consists of four phases: the setup phase, the registration phase, and the authentication phase. The notations used in this paper are shown in Table 1.

Table 1: Notions and their explanations

Notations	Explanations
U	The remote user
S	The remote server
X → Y:M	X sends a message M to Y
username	The identity of user U
PW	The password of user U
$E_p(a,b)$	An elliptic curve equation with order n
s	The long-live secret key of server S
$P_pub = sP$	The long-live public key of server S
SK	A session key
$h(\cdot)$,$h_1(\cdot)$,$h_2(\cdot)$	Three secure one-way hash functions
Z_q^*	Multiplication group of Z_q
$\|$	The string concatenation operator
$E_s(\bullet)$	Symmetric key encryption under the key s

3.1 System Setup Phase

Step 1. The server selects an elliptic curve equation $E_p(a,b)$ with the order n, and chooses a base point P over $(E)_p(a,b)$, where n is a large number for the security consideration. Then, it chooses a random number $s \in_R (Z_p)^*$ as the secret key and computes the public key $(P)_{pub} = sP$;

Step 2. The server selects three one-way hash functions, $h(\cdot) : \{0,1\}^* \to \{0,1\}^k$, $h_1(\cdot) : G \times \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^k$, $h_2(\cdot) : G \times G \times \{0,1\}^* \to \{0,1\}^k$,where G is a cyclic addition group generated by P over $(E)_p(a,b)$;

Step 3. The server publishes $\{E_p(a,b)$, P, P_{pub}, $h(\cdot)$, $h_1(\cdot)$, $h_2(\cdot)\}$ and keeps s in secret.

3.2 Registration Phase

In this phase, the user registers on the SIP server through a secure channel. When a user wants to login into the remote server, he/she firstly should register to the remote server..The details of this phase are as follows.

R1: The user U selects his or her username, password PW and a random number $a \in_R Z_p^*$. After that, U computes $h(PW\|a)$ and sends $\{h(PW\|a), username\}$ to the server through a secure channel.

R2: after receiving the registration information, the server computes $R = h(h(PW\|a)\|username)s(-1)P$ and $X = h(username\|s)P$. Then the server stores R and X into the smart card and issues it to U.

R3: Upon receiving the card, U stores a in the card. Then the card contains (R, X, a).

3.3 Authentication Phase

Whenever the user wants to login into the remote server, he/she performs the following.

A1: $U \to S : REQUEST(username, V, W)$ U inserts his/her smart card into a card reader and inputs his/her username and password PW. Then, U's smart card picks a random number $b \in_R (Z_p)^*$, and computes V= $bR + X$ and $W = h(h(PW\|a)\|username)P_{pub}$. Next, the card sends a request message $REQUEST(username, V, W)$ to the server.

A2: $S \to U : CHALLENGE(realm, Auth_s, S, r)$ After receiving the request message, the server S computes $(X)' = h(username\|s)P$ and $W' = s^2(V - X')$. Then, it checks $W \stackrel{?}{=} W'$, if true it chooses two random integers $c \in_R Z_p^*$ and $r \in_R Z_p^*$. Then computes $S = cP$, $K = cs(V - X')P$, $SK = h_1(K\|r\|username)$ and $Auth_s = h_2(K\|W\|r\|SK)$. Next, it sends message $CHALLENGE(realm, Auth_s, S, r)$ to U over a public channel.

A3: $U \to S : RESPONSE(realm, Auth_u)$ Upon receiving message $CHALLENGE(realm, Auth_s, S, r)$, U computes $K = bh(h(PW\|a)\|username)S$ and $SK = h_1(K\|r\|username)$ and verifies if $Auth_s \stackrel{?}{=} h_2(K\|h(h(PW\|a)\|username)bP_{pub}\|r\|SK)$.
If so, U computes $Auth_u = h_2(K \| h(h(PW \| a) \| username) bP_{pub} \| r + 1\|SK)$ and sends $RESPONSE(realm, Auth_u)$ back to the server over public channel. Otherwise, it deletes received information and the protocol stops.

A4: After receiving the RESPONSE message, the server verifies $Auth_u \stackrel{?}{=} h_2(K\|W'\|r + 1\|SK)$. If the message is authenticated, the server sets SK a shared session key with user U. Otherwise, it deletes received information and the protocol stops.

3.4 Password Changing Phase

This phase is similar to the Zhang et al.'s password changing phase. When the user U wants to update its password, it needs to agree on a session key with the server via the authentication phase in advance.

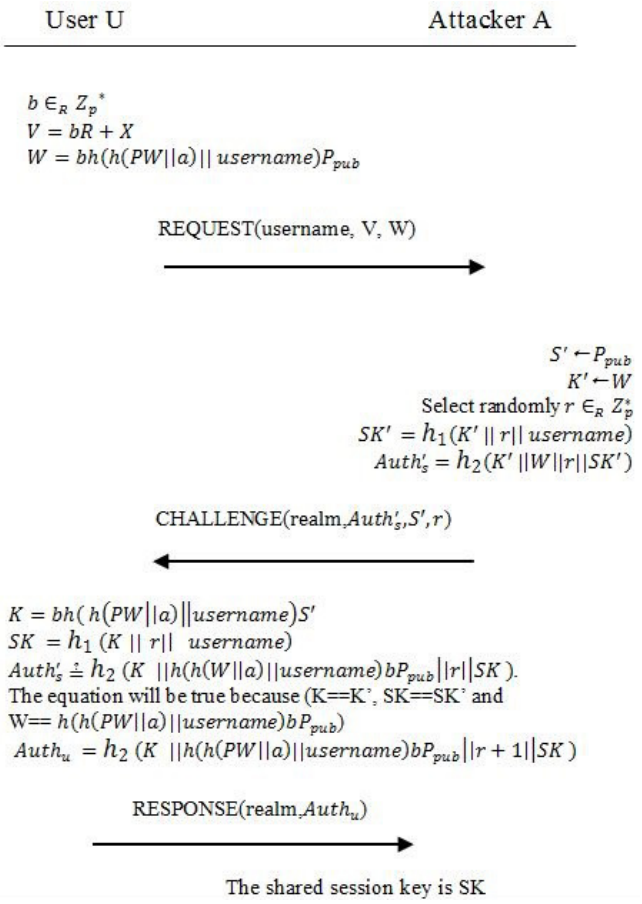

Figure 2: Server spoofing attack on Jiang et al.'s scheme

4 Cryptanalysis of Jiang et al.'s Scheme

Jiang et al. claimed that their protocols can resist various attacks. However, in this section, we will show that the Server spoofing attack, not as they claimed, is still effective in Jiang et al.'s protocol.

Let A be an attacker. A can eavesdrops the message REQUESTusername,V,W transmitted between server S and user U. A can get server′s public key, because S has published it with other parameters. Then, A can execute server spoofing attack. The detail of attack is illustrated in Figure 2 and is presented as follows.

Step 1. U inputs his username and password PW after inserting his smart card in card reader. The card generates randomly a number $b \in_R Z_p^*$ and computes $V = bR + X$ and $W = bh(h(PW\|a), username)P_{pub}$. Then, the card sends a request message REQUEST($username$, V, W) to S.

Step 2. A eavesdrops message REQUEST($username$, V, W) and get $username$, V, W. he/she generates a random number $r \in_R Z_p^*$. Next he/she get server public key, put its value in $S'(S' \leftarrow P_{pub})$ and put value of W in $K'(K' \leftarrow W)$. Then he/she computes $SK' = h_1(K'\|rl|username)$ and $Auth'_s = h_2(K'\|W\|r\|SK')$. Next, A sends message CHALLENGE(realm, $Auth'_s, S', r$) to U.

Step 3. Upon receiving message CHALLENGE (realm, $Auth'_s, S', r$), U computes $K = bh(h(PW \| a) \| username)S'$ and $SK = h_1(K \| r \| username)$ and verifies if $Auth'_s \stackrel{?}{=} h_2(K\|h(h(PW\|a)\|username)bP_{pub}\|r\|SK)$. The user will find true because:

$$\begin{aligned} K &= bh(h(PW\|a)\|username)S' \\ &= bh(h(PW\|a)\|username)P_{pub} \\ &= W \\ &= K' \\ SK &= h_1(K\|r\|username) \\ &= h_1(K'\|r\|username) \\ &= SK' \\ W &= bh(h(PW\|a)\|username)P_{pub} \\ &= h(h(PW\|a)\|username)bP_{pub}. \end{aligned}$$

Then user U authenticates attacker A and sends to him RESPONSE thinking that he/she communicate with a legal server S.

According to previous analysis, the adversary can easily impersonate identity of server at any time. The user U does not know whether the one he contacts is that the valid server or not. So the adversary can impersonate the server successfully. Therefore, Jiang et al.'s protocol is vulnerable to the server spoofing attack.

5 Our Proposed Protocol

In this section, in order to overcome weakness in Jiang et al.s protocol, we propose an improved and efficient authentication and key agreement protocol for SIP. Our protocol consists of four phases, which are system setup phase, registration phase, authentication and key agreement phase, and password changing phase. These phases are described as follows.

5.1 System Setup Phase

In this section, the server selects an elliptic curve equation $E_p(a, b)$, over a finite field F_q, an additive group G of order p and P a base point generator with order n over equation $E_p(a, b)$, n is a large prime of height entropy. Then, the server picks a random integer $s \in_R Z_p^*$ as its secrete key, and computes its public key $P_{pub} = sP$. Next, the server chooses three one-way hash functions $h(\cdot)$, $h_1(\cdot)$ and $h_2(\cdot)$. Finally, the server publishes all parameters except its private key, which it is saved secretly.

5.2 Registration Phase

When user wants to register in server and become a legal user, he has to perform the following steps.

R1: The user U selects his or her username, password PW and a random number $a \in_R Z_p^*$. After that, U computes $h(PW\|a)$ and sends $\{h(PW\|a), username\}$ to the server over a secure channel.

R2: after receiving the registration information, the server computes $R = h(h(PW\|a)\|username)s^{-1}P$ and $X = h(username\|s)P$. Then, the server stores R and X into the smart card and issues it to U.

R3: Upon receiving the card, U stores a in the card. Therefore, user card contains (R,X,a).

5.3 Authentication and Key Agreement Phase

As illustrated in Figure 3, whenever a legal user U wishes to log into the server, he/she have to inserts his/her smart card in card reader and inputs his/her username and password PW. Next, the following steps will be executed between server S and user U.

Auth 1: $U \rightarrow S$:$REQUEST(username, V, W)$

After inserting the smart card in card reader and inputting the username and password; the smart card of user U chooses a random $b \in_R Z_p^*$, and computes $V = bR + X$, $Y = bh(h(PW\|a), username)$ and $W = YP_{pub}$, then, he/she sends a request message $REQUEST(username, V, W)$ to the server over a public channel.

Auth 2: $S \rightarrow U$: $CHALLENGE(realm, Auth_s, S, r)$

When server S gets the request message, it computes $X' = h(username\|s)P$ and $W' = s^2(V - X')$. Then, it verifies $W \stackrel{?}{=} W'$. If true, U is authenticated and the server S picks randomly two integers $c, r \in_R Z_p^*$. Then, it computes $S = cP$, $K = cs(V - X')$, $SK = h_1(K\|r\|username\|X')$ and $Auth_s = h_2(K\|W\|r\|SK\|X')$. Next, it sends message $CHALLENGE(realm, Auth_s, S, r)$ to U over a public channel.

Auth 3: $U \rightarrow S$:$RESPONSE(realm, Auth_u$

Once the user U receives the CHALLENGE message, it calculates $K = YS$ and $SK = h_1(K\|r\|username\|X)$. Then, checks $h_2(K\|W\|r\|SK\|X)$ if is true, the server is authenticated. Then, user U computes $Auth_u$ as following $Auth_u = h_2(K\|W\|r + 1\|SK\|X)$ and sends $RESPONSE(realm, Auth_u)$ back to the server over public channel. Otherwise, it stops the protocol and deletes received and calculated parameters.

Auth 4: After receiving the RESPONSE message, the server computes $h_2(K\|W'\|r + 1\|SK\|X')$ and verifies that it equal to received $Auth_u$. If successful, the server sets SK a shared session key with user U. Otherwise, it stops the protocol and deletes received and calculated parameters

5.4 Password Changing Phase

This phase is similar to Zhang et al.s password changing phase. When the user U wants to update its password, it needs to agree on a session key with the server via the authentication phase in advance. The details of this phase are described as following.

Pass 1. $U \rightarrow S$: $(username, e, New_u)$

The user U chooses its new password PW^* and two random integers a^*, $e \in_R Z_p^*$ and computes $h(PW^*\|a^*)$ and $tag_u = h(username\|e\|h(PW^*\|a^*))$, it then uses SK to encrypt the new parameters: $New_u = E_{KS}(username\|e\|h(PW^*\|a^*)\|tag_u)$. Next, it sends message$(username, e, New_u)$ to server.

Pass 2. $S \rightarrow U$: (New_s)

Upon receiving the information, the server decrypts the message and then checks the validity of the authentication $tag_u \stackrel{?}{=} h(username\|e\|h(PW^*\|a^*))$. If it is valid, the server computes the new secret information $R^* = h(h(PW^*\|a^*)\|username)s^{-1}P$ and $tag_s = h(username\|e + 1\|R^*)$. Then, it sends encryption information $New_s = E_{KS}(R^*\|tag_s)$ to the user U.

Pass 3. The user U decrypts received message and verifies the validity of $tag_s \stackrel{?}{=} h(username\|e + 1\|R^*)$. If it is valid, the user U stores R^* and a^* in its smart card.

6 Security Analysis

In this section we will prove that our protocol provide mutual authentication and session key secrecy. Moreover, we will show that its secure against several attacks especially server spoofing attack, user impersonation attack, Denning-Sacco attack, replay attack, stolen verifier attack, offline password guessing attack, and man-in-the-middle attack.

6.1 Mutual Authentication

Mutual authentication means that both the user and server are authenticated to each other within the same protocol. In the proposed scheme the server can authenticate user after receiving REQUEST by checking W, and after receiving RESPONSE by checking $Auth_u$. Upon receiving message CHALLENGE user can authenticate the server by testing validity of $Auth_s$. Consequently, the proposed protocol provides mutual authentication.

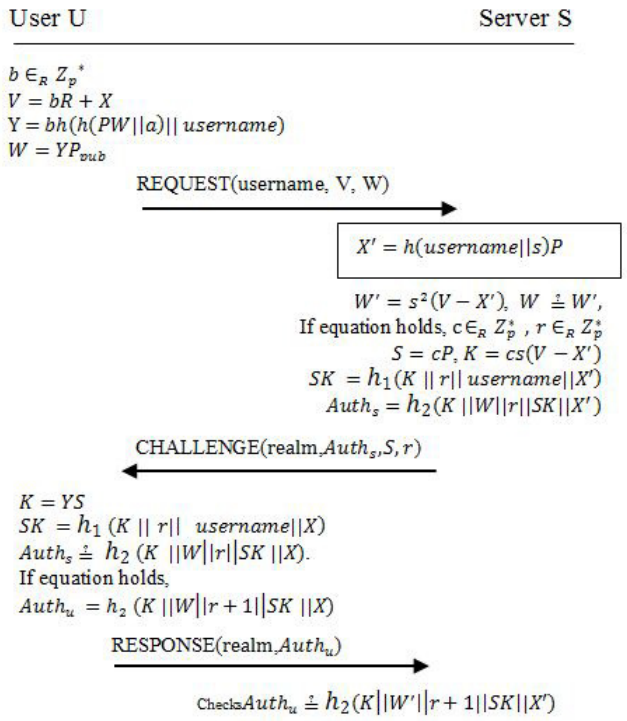

Figure 3: Authentication phase of our proposed scheme

6.2 Session Key Secrecy

Session key security means that at the end of the key exchange anyone cannot know the session key excepting the legal communication parties (the user and the server). In the proposed scheme the session key is computed in this way $SK = h_1(K\|r\|username\|X)$ where $X' = h(username\|s)P$ and $K = bh(h(PW\|a)\|username)S$. Since, PW, a and s are secret the session key cannot calculate by anyone except the server and the client. Therefore, our proposed protocol provides session key secrecy.

6.3 Server Spoofing Attack

The proposed scheme can resist against server spoofing attack. Assume that attacker Alice wants to impersonate the server and spoof user U, Alice has to computes $Auth_s = h_2(K\|W'\|r\|S\|K\|X')$. However, Alice does not have any information about a server secret key s. Then, she cannot compute K,SK and X'. Therefore, Alice cannot forge a valid CHALLENGE message.

6.4 User Impersonation Attack

Assume that attacker Alice wishes to connect to the server as legitimate user U. Alice has to prove its validity by forging two messages $REQUEST(username, V, W)$ and $RESPONSE(realm, Auth_u)$. While Alice need to know some secret information PW,a and X. Therefore, Alice is not capable to send the two validate messages. As result, our scheme can resist user impersonation attack.

6.5 Denning-Sacco Attack

The Denning-Sacco attack is when User or Server 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.

In our scheme, the session key is calculated in this way $SK = h_1(K\|r\|username\|X)$ or $SK = h_1(K\|r\|username\|X')$. If an attacker obtains a session key, he will have to break the one-way hash function to get K,r and X or X. Then he have to know a secrete a and face the ECDLP if he want to guess password (PW') and verify the validity of $Auth'_s \stackrel{?}{=} h_2(K\|h(h(PW'\|a)\|username)bP_{pub}\|r\|SK\|X)$. So, the proposed scheme is secure against Denning Sacco attack.

6.6 Replay Attack

A replay attack is applied when an adversary reuse the information obtained in a protocol, trying to impersonate or deceive another legitimate participant. The following explain why the proposed protocol can resist to this attack. The adversary Alice may intercept the messages $REQUEST(username, V, W)$ and $RESPONSE(realm, Auth_u)$ from the User U and try to impersonate a legitimate user. However, she cannot calculate V, W and $Auth_u$ since she don't know server secret key. Alice has to face the ECDLP, if she wants get the correct one by guessing the secret key s from V or W. after replaying REQUEST or RESPONSE the server will detect the attack via comparing if $W \stackrel{?}{=} s^2(V - X')$ or $Auth_u \stackrel{?}{=} h_2(K\|W'\|r + 1\|SK\|X')$. Now, Suppose that Alive intercepts the message $CHALLENGE(realm, Auth_s, S, r)$ and try to replays it to impersonate the legal server. In order, to be authenticated by the user, Alice have to computes the value of $h_2(K\|W\|r\|SK\|X)$ using secret PW,X,a, $K = YS$ and $SK = h_1(K\|r\|username\|X)$. Since Alice don't have information about secret parameters she cannot computes a valid $Auth_s$. As result the proposed protocol withstand replay attack.

6.7 Stolen Verifier Attack

The stolen verifier attack means that an adversary steals the secret information from the server, like user's password. Then, the adversary uses it directly to masquerade as a legitimate user in a user authentication connection.

In the proposed scheme, any user's secret is stored in server database, so the attackers cannot obtain the user's secret information from server. Therefore, our proposed protocol is secure against stolen verifier attack.

6.8 Offline Password Guessing Attack

Password guessing attacks means that when an attacker interposes the communication between user and server

then he can guess the correct secret password by repeatedly guessing possible passwords and verifying the correctness of the guesses.

Suppose an attacker records all messages (REQUEST, CHALLENGE and RESPONSE) transmitted between user and server, then extract $username$, V, W, $realm$, $Auth_s$, S, r and $Auth_u$, and tries to guess the password PW^* and verifies its correctness. Since the attacker does not know any information about values of s, a and b he cant compute K, X, SK and $h(h(PW\|a)\|username)bP_{pub}$. Then, he cant verify the calculated V, W, $Auth_s$ or $Auth_u$.

If attacker steals user card he can get R, a and X, he must to know s to checks $h(h(PW^*\|a)\|username)s^{-1}P$. However, he will face ECDLP to extract s from $X = h(username\|s)P$. Therefore, our proposed scheme is safe against password guessing attack.

6.9 Man-in-the-Middle Attack

Man-in-the-middle attacks means that the attacker makes independent connections with the victims and relays messages between them, making them believe that they are talking directly to each other over a private connection. However, the entire conversation is controlled by the attacker.

In our protocol all messages are authenticated by server or user, to know their origin. In addition, at the end of authentication, the session key is shared between user and server, so the following messages will be encrypt using session key. To replay these messages, an attacker needs to know a session key. But, he cannot calculate it since he does not know s, a, X, PW and b. As result, our protocol is secure against Man-in-the-middle attack.

7 Performance Comparison

In this section, the performance of our proposed authentication and key agreement schemes is compared with other related authentication protocols. In this comparison a very lightweight operations like string concatenation operation, Exclusive-OR operation are not examined, because there computation cost is negligible. The notations used are illustrated as follows.

- T_h: Computational cost of one-way hash operation.
- T_{pm}: Computational cost of elliptic curve point multiplication.
- T_{pa}: Computational cost of elliptic curve point addition.
- T_{inv}: Computational cost of modular inversion.
- T_{EKs}: Computational cost of symmetric encryption.
- T_{DKs}: Computational cost of symmetric decryption.

In the registration phase of our protocol the user uses one hash function and the server computes $2T_h + 2T_{pm} + 1T_{inv}$. When the user need to be authenticated by server, it calculates $5T_h + 3T_{pm} + 1T_{pa}$ and the server computes $4T_h + 4T_{pm} + 1T_{pa}$. In the password changing phase the user computes $3T_h + 1T_{EKs} + 1T_{DKs}$ and the server computes $3T_h + 1T_{pm} + 1T_{inv} + 1T_{EKs} + 1T_{DKs}$.

In Table 2, we have illustrated the security performance of related schemes, as we can show ours protocol is secure against stolen verifier attack, Denning-Sacco attack, off-line password guessing attack, replay attack, man in the middle attack, server spoofing attack, insider attack, impersonation attack and denial of service attack. But, the Jiang et al. protocol is not secured against Server Spoofing attack and suffer from impersonation attack, and don′t provide security against man-in-the middle attack and Denning Sacco attack. So, we can say that our protocol is more secured if it is compared with Jiang et al.'s scheme.

According to Table 3, we can observe that our protocol reduce the number of T_{pm} from 4 to 3 in the authentication phase, if it's comparede with the same phase of Jiang et al. protocol. Hence, we can say that authentication phase of our protocol is faster than the same phase of Jiang et al.'s protocol. So, our protocol is more efficient than Jiang et al.'s protocol.

Table 2: Security comparison

Attacks	Zhang et al.	Tu et al.	Jiang et al.	Lin et al. [8]	Ours
Stolen Verifier	Yes	Yes	Yes	Yes	Yes
Denning Sacco	Yes	-	-	-	Yes
Password Guessing	Yes	Yes	Yes	Yes	Yes
replay	Yes	Yes	Yes	Yes	Yes
Man in the Middle	Yes	No	-	-	Yes
Server Spoofing	-	No	No	Yes	Yes
Impersonation	No	No	No	Yes	Yes
Mutual Authentication	Yes	Yes	Yes	Yes	Yes
Session Key Secrecy	Yes	-	Yes	Yes	Yes

8 Conclusion

In this article, we demonstrated that the protocol proposed by Jiang et al. cannot withstand server spoofing attacks. In order to overcome this weakness we proposed an efficient and secure SIP authentication scheme. By

Table 3: Computational comparisons between our protocol and related protocols

Phases	Entities	Zhang et al.	Tu et al.	Jiang et al.	Lin et al.	Ours
Registration Phase	User	$1T_h$	$1T_h$	$1T_h$	$1T_h$	$1T_h$
	Server	$1T_h + 1T_{pm} + 1T_inv$	$1T_h + 1T_{pm}$	$2T_h + 2T_{pm} + 1T_inv$	$2T_h + 1T_{Eks}$	$2T_h + 2T_{pm} + 1T_{inv}$
Authentication Phase	User	$6T_h + 4T_{pm} + 1T_{pa}$	$5T_h + 4T_{pm} + 1T_{pa}$	$4T_{pm}$	$6T_h + 3T_{pm} + 1T_{EKs} + 1T_{DKs}$	$5T_h + 3T_{pm} + 1T_{pa}$
	Server	$4T_h + 4T_{pm} + 1T_{pa}$	$5T_h + 3T_{pm}$	$4T_h + 4T_{pm} + 1T_{pa}$	$5T_h + 3T_{pm} + 2T_{EKs} + 2T_{DKs}$	$4T_h + 4T_{pm} + 1T_{pa}$
Password Changing Phase	User	$3T_h + 1T_{EKs} + 1T_{DKs}$	$3T_h + 1T_{EKs} + 1T_{DKs}$	–	$6T_h + 3T_{pm} + 1T_{EKs} + 1T_{DKs}$	$3T_h + 1T_{EKs} + 1T_{DKs}$
	Server	$3T_h + 1T_{pm} + 1T_{EKs} + 1T_{DKs} + 1T_{inv}$	$3T_h + 1T_{pm} + 1T_{EKs} + 1T_{DKs}$	–	$5T_h + 3T_{pm} + 2T_{EKs} + 2T_{DKs}$	$3T_h + 1T_{pm} + 1T_{EKs} + 1T_{DKs} + 1T_{inv}$
Total		$18T_h + 10T_{pm} + 2T_{pa} + 2T_{inv} + 2T_{EKs} + 2T_{DKs}$	$18T_h + 9T_{pm} + 1T_pa + 2T_{EKs} + 2T_{DKs}$	$12T_h + 10T_{pm} + 2T_{pa} + 1T_{inv}$	$25T_h + 12T_{pm} + 7T_{EKs} + 6T_{DKs}$	$18T_h + 10T_{pm} + 2T_{pa} + 2T_{inv} + 2T_{EKs} + 2T_{DKs}$

analyzing our scheme, we show that it is secure against various attacks and can provide many security services. Then, we conclude that our proposed protocol is suitable for Telephony over IP applications

References

[1] R. Arshad and N. Ikram, "Elliptic curve cryptography based mutual authentication scheme for session imitation protocol," *Multimedia Tools Appl*, vol. 66, no. 2, pp. 165–178, 2013.

[2] M. Azrour, M. Ouanan, and Y. Farhaoui, "SIP authentication protocols based on elliptic curve cryptography: Survey and comparison," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 4, no. 1, pp. 231–239, 2016.

[3] W. Diffie and M. Hellman, "New directions in cryptology," *IEEE Transaction on Information Theory*, vol. 22, no. 6, pp. 644–654, 1976.

[4] A. Durlanik and I. Sogukpinar, "Sip authentication scheme using ecdh," *World Enformatika Socity Transations on Engineering Computing and Technology*, vol. 8, pp. 350–353, 2005.

[5] J. Franks, P. Hallam-Baker, J. Hostetler, S. Lawrence, P. Leac, A. Luotonen, and L. Stewart, "Http authentication: Basic and digest access authentication," Tech. Rep. RFC 2617, June 1999.

[6] M. Handley, H. Schulzrinne, E. Schooler, and J. Rosenberg, "SIP: Session initiation protocol," Tech. Rep. RFC 2543, Mar. 1999.

[7] H.H. Hilinc, Y. Allaberdiyev, and T. Yanik, "Efficient ID-based authentication and key argeement protocols for the session initiation protocol," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 23, pp. 560–579, 2015.

[8] H.Lin, F. Wen, and C.Du, "An anonymous and secure authentication and key agreement scheme for session initiation protocol," *Multimed Tools Appl*, vol. DOI 10.1007/s11042-015-3220-2, 2016.

[9] H. Huang, W.Wei, and G. E. Brown, "A new efficient authentication scheme for session initiation protocol," in *The 9th Joint Conference on Information Sciences*, 2006.

[10] A. Irshad, M. Sher, E. Rehman, ChS. Ashraf, MU. Hassan, and A. Ghani, "A single round-trip sip authentication scheme for voice over internet protocol using smart card," *Multimed Tools Applications*, 2013.

[11] Q. Jiang, J. Ma, and Y. Tian, "Cryptanalysis of smart-card-based password authenticated key agreement protocol for session initiation protocol of zhang et al," *International Journal of Communication Systems*, vol. 28, no. 7, 2014.

[12] H. Jo, Y. Lee, M. Kim, S. Kim, and D. Won, "Offline password guessing attack to yangs and huangs authentication schemes for session initiation protocol," in *The 5th International Joint Conference on INC, IMS and IDC (NCM'09)*, pp. 618–621, Aug. 2009.

[13] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of Computation*, vol. 48, no. 177, pp. 203–209, 1987.

[14] F. W. Liu and H. Koenig, "Cryptanalysis of a sip authentication scheme," in *12th IFIP TC6/TC11 International Conference (CMS'11)*, pp. 134–143, 2011.

[15] J. Lopez and R. Dahab, "An overview of elliptic curve cryptography," Tech. Rep., June 2000.

[16] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, "Sip: Session initiation protocol," Tech. Rep. RFC 3261, June 2002.

[17] M. Stanek, "Weaknesses of password authentication scheme based on geometric hashing," *International Journal of Network Security*, vol. 18, no. 4, pp. 798–801, 2016.

[18] H. Tang and X. Liu, "Cryptanalysis of arshad et al.'s ecc-based mutual authentication scheme for session initiation protocol," *Multimedia Tools and Applications*, vol. 65, no. 3, pp. 165–178, 2013.

[19] 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.

[20] H. Tu, N. Kumar, N. Chilamkurti, and S. Rho, "An improved authentication protocol for session initiation protocol using smart card," *Peer-to-Peer Networking and Applications*, vol. 8, no. 5, pp. 903–910, 2014.

[21] Y. Wang and X. Peng, "Cryptanalysis of two efficient password-based authentication schemes using smart cards," *International Journal of Network Security*, vol. 17, no. 6, pp. 728–735, 2015.

[22] J. Wei, W. Liu, and X. Hu, "Secure and efficient smart card based remote user password authentication scheme," *International Journal of Network Security*, vol. 18, no. 4, pp. 782–792, 2016.

[23] L. Wu, Y. Zhang, and F. Wang, "A new provably secure authentication and key agreement protocol for sip using ecc," *Computer Standards and Interfaces*, vol. 31, no. 2, pp. 286–291, 2009.

[24] Q. Xie, "A new authenticated key agreement for session initiation protocol," *International Journal of Communication Systems*, vol. 25, no. 1, pp. 47–54, 2012.

[25] 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.

[26] E. J. Yoon and K. Y. Yoo, "Cryptanalysis of nake protocol based on ecc for sip and its improvement," in *Second International Conference on Future Generation Communication and Networking Symposia*, 2008.

[27] E. J. Yoon and K. Y. Yoo, "Cryptanalysis of ds-sip authentication scheme using ecdh," in *International Conference on New Trends in Information and Service Science*, pp. 642–647, Aug. 2009.

[28] E. J. Yoon and K. Y. Yoo, "A new authentication scheme for session initiation protocol," in *International Conference on Complex, Intelligent and Software Intensive Systems (CISIS'09)*, pp. 549–554, 2009.

[29] E. J. Yoon, K. Y. Yoo, C. Kim, Y. S. Hong, M. Jo, and H. H. Chen, "A secure and efficient sip authentication scheme for converged voip networks," *Computer Communications*, vol. 33, no. 14, pp. 1674–1681, 2010.

[30] L. Zhang, S. Tang, and Z. Cai, "Efficient and flexible passwordauthenticated key agreement for voice over internet protocolsession initiation protocol using smart card," *International Journal of Communication Systems*, vol. 27, no. 11, p. 2691–2702, 2013.

[31] L. Zhang, S. Tang, and Z. Cai, "Cryptanalysis and improvement of password-authenticated key agreement for session initiation protocol using smart cards," *Secur Communication Networks*, 2014.

Biography

Mourade Azrour received his Master's degree in Computer Science and Distributed Systems in 2014 from Departments of Mathematics and Computer Science, Faculty of Science, University Ibn Zohr, Agadir, Morocco. He is currently a Ph.D. candidate of the Moulay Ismail University, Faculty of sciences and Techniques, Errachidia, Morocco.His research interests include Authentication Protocols, Computer and Network Security and Cryptography.

Yousef Farhaoui is an professor, Department of Computer Science in Faculty of sciences and Techniques, Moulay Ismail University, Morocco. Received his PhD degree in computer security from the University Ibn Zohr. His research interest includes computer security, Data Mining, Data Warehousing, Data Fusion etc..

Mohammed Ouanan was born in Morocco on 1971. He received the B.S. degree in applied Mathmatics and the PhD degree in applied Mathmatics, all from the Faculty of science, University of Fez, Morroco, in Morocco, in 1997, 1999, and 2002 respectively. Since JULAY 2007, he has been an Assistant Professor with the Department of Computer Sciences, Faculty of Sciences and Techniques Errachidia, Moulay Ismail University, Meknes, Morocco. His research interest includes security, indexed image, statistical signal processing, multidimensional signal modelling, and Mathmatics.