Authentic Caller: Self-Enforcing Authentication in a Next-Generation Network

Muhammad Ajmal Azad, Samiran Bag, Charith Perera, Mahmoud Barhamgi, and Feng Hao

***Abstract*—The Internet of Things (IoT) or the cyber-physical system (CPS) is the network of connected devices, things, and people that collect and exchange information using the emerging telecommunication networks (4G, 5G IP-based LTE). These emerging telecommunication networks can also be used to transfer critical information between the source and destination, informing the control system about the outage in the electrical grid, or providing information about the emergency at the national express highway. This sensitive information requires authorization and authentication of source and destination involved in the communication. To protect the network from unauthorized access and to provide authentication, the telecommunication operators have to adopt the mechanism for seamless verification and authorization of parties involved in the communication. Currently, the next-generation telecommunication networks use a digest-based authentication mechanism, where the call-processing engine of the telecommunication operator initiates the challenge to the request-initiating client or caller, which is being solved by the client to prove his credentials. However, the digest-based authentication mechanisms are vulnerable to many forms of known attacks, e.g., the man-in-the-middle (MITM) attack and the password guessing attack. Furthermore, the digest-based systems require extensive processing overheads. Several public-key infrastructure (PKI)-based and identity-based schemes have been proposed for the authentication and key agreements. However, these schemes generally require a smart card to hold long-term private keys and authentication credentials. In this article, we propose a novel self-enforcing authentication protocol for the session-initiation-protocol-based next-generation network, based on a low-entropy shared password without relying on any PKI or the trusted third party system. The proposed system shows effective resistance against various attacks, e.g., MITM, replay attack, password guessing attack, etc. We analyze the security properties of the proposed scheme in comparison to the state of the art.**

***Index Terms*—Authorization, identity spoofing, password-based authentication, session-initiation-protocol (SIP) authentication, self-enforcing authentication.**

I. Introduction

OVER the last few decades, advances in networking technologies, communication systems, improved processing power, and availability of new tools, applications, and software have changed the way Internet-connected devices, people, smart systems communicate and exchange information with minimal human involvement [1]. The Internet of Things (IoT) and cyber-physical system (CPS) are the major driving forces in the smart interconnected environment. Though the smart-connected environment has brought a lot of benefits to the humanity but its success depends on the security, privacy [2], and trust of the stakeholders (in particular, users of the IoT devices) involve in the connected world.

Within this connected scenario, it is utmost that sensitive information should only be originated and communicated from the authorized participants. The information from the compromised devices and people would bring detrimental consequences to the network.

The emerging telecommunication technologies [4G, 5G, IP-based cores, i.e., voice over IP (VoIP), long term evolution (LTE) and IP multimedia subsystems (IMS)] are the main communication technologies used by the IoT and CPS systems for transmitting time-critical and sensitive information between the monitored source and the centralized processing unit. Today, telecommunication systems are also used to confirm some of the most sensitive transactions, e.g., two-factor authentication for the code and identity verification, the one-time passcode for the bank transactions, proving the identity in the event of a disaster, and reporting sensitive information between the entities (e.g., from the electrical grid to control systems). The emerging telecommunication networks [IMS, LTE, and next-generation network (NGN) and IP-based networks (VoIP)] have adopted session initiation protocol (SIP) for the creation, modification, termination, and management of the communication session between the participants (e.g., source and destination). The SIP management messages are similar to the HTTP message and are text based [3]. The SIP-based networks consist of two major components: the SIP user agent (UA) and the SIP network server (NS). The SIP UA is the end user responsible for initiating and accepting the connection. The SIP NS provides the bridge for establishing a connection between the source and destination.

Manuscript received June 27, 2019; revised August 6, 2019; accepted September 2, 2019. Date of publication September 19, 2019; date of current version February 6, 2020. The work of F. Hao and S. Bag was supported in part by the ERC Starting Grant 306994 and in part by the Royal Society Grant ICA/R1/180226. Paper no. TII-19-2741. *(Corresponding author: Muhammad Ajmal Azad.)*

M. A. Azad is with the Department of Computer Science, The University of Derby, DE22 1 GB Derby, U. K. (e-mail: m.azad@derby.ac.uk).

S. Bag and F. Hao are with the Department of Computer Science, The University of Warwick, CV4 7AL Coventry, U.K. (e-mail: samiran.bag@warwick.ac.uk; feng.hao@warwick.ac.uk).

C. Perera is with the School of Computer Science and Informatics, Cardiff University, CF10 3AT Cardiff, U.K. (e-mail: charith.perera@ieee.org).

M. Barhamgi is with the Claude Bernard Lyon 1 University, 69100 Villeurbanne, France (e-mail: mahmoud.barhamgi@univ-lyon1.fr).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TII.2019.2941724

The openness of IP-based networks makes emerging networks vulnerable to many security threats, e.g., denial of service attacks, authentication attacks, and misuse of the telephone system for the unwanted communications [4]–[7]. Authenticating users in these networks is very important for user security and the reliable communication of sensitive information over the networks. The first thing for reliable communication is establishing trust on the identities owned by the participants. The original SIP protocol uses the HTTP digest authentication protocol for authenticating the users in the network. In digest authentication, the proxy server initiates a challenge to the call initiator, and the call initiator solves the challenge to prove his credentials. However, the HTTP digest authentication not only has a high computational cost and communication overheads but also does not provide effective security [7]–[9] under many attacks. For example, digest authentication does not provide mutual authentication, does not provide complete message integrity, and is also vulnerable to the password guessing attack. The security of the digest authentication can be improved by adding SSL/TLS to SIP messages but it requires trusted authorities for the management of certificates. Several public-key cryptography [10]–[13] methods have also been proposed for the authentication but these systems require a public-key infrastructure (PKI) to distribute the public keys between the client and the proxy servers. A number of password-based authentication solutions [14], [15] have also been proposed, but many of these are found to be insecure. For example, [14] and [15] is vulnerable to an offline password-guessing attack and [16] is subject to compromise of old session keys (Denning–Sacco attack).

The authentication mechanism of the SIP should be efficient (small communication and computation overhead) and secure against a number of security attacks. Developing a cryptographic authentication system for the SIP protocol without the PKI with inherent properties of effective resistance against attacks is indeed a challenging task. To provide an efficient authentication mechanism without any PKI, in this article, we propose a new password-based self-enforcing authentication scheme for user/client authentication in an NGN. The scheme enables the proxy server and the SIP clients to exchange their authentication information over an open and insecure network based on a password without requiring any PKI. Our scheme ensures several security properties even under a strong adversary with the use of low entropy password. The new authentication scheme provides effective security against different types of attacks and strong adversaries, e.g., replay attack, man in the middle attack, password guessing attack, etc. We comprehensively analyze the security properties of the proposed scheme and compare them with the state of the art. Furthermore, we prototype the protocol and analyze its performance for computation and communication overheads. The results show that the scheme does not incur high bandwidth and computation overheads.

The rest of this article is organized as follows. Section II presents an overview of the proposed scheme followed by a comprehensive discussion on the security properties of the scheme in Section III. Section IV provides complexity analysis.

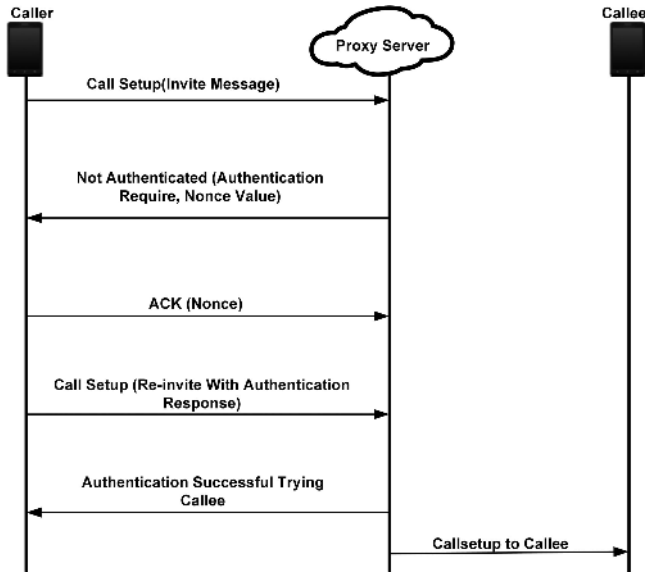

Fig. 1. Authentication mechanism in the NGN.

Section V reviews the existing authentication mechanisms. Finally Section VI concludes this article.

II. Authentic Call: Self-Enforcing Authentication in Modern NGNs

In this article, we aim to explore a lightweight cryptographic solution to authenticate the client in a next-generation telecommunication network without requiring any PKI. The new authentication scheme allows the proxy server to authenticate the SIP client based on a shared low-entropy password and the authentication process remains consistent within the message structure of SIP RFC-3261 [3].

A. Authentication in NGNs

Authentication process provides a mechanism to verify that a caller or the callee possess the credentials he claims. In the NGN, the SIP protocol uses a challenge-response-based authentication process for authenticating the end user. It is similar to the digest authentication as used in the HTTP protocol and employs an MD5 hash algorithm to encode the user credentials (username, realm, password, and digest URI). The building block of the SIP authentication process is shown in Fig. 1. The proxy server or the call processing engine on receiving the call request or registration request initiates a challenge to the caller, which he has to solve correctly in order to authenticate and associate himself with the proxy server.

B. Overview of Self-Enforcing Authentication Scheme

In a password-authenticated key exchange (PAKE) scheme, two or more parties (between a client and a server or between two clients) authenticate themselves to each other based on their knowledge of a password. The parties establish a cryptographic session key by exchanging a series of messages

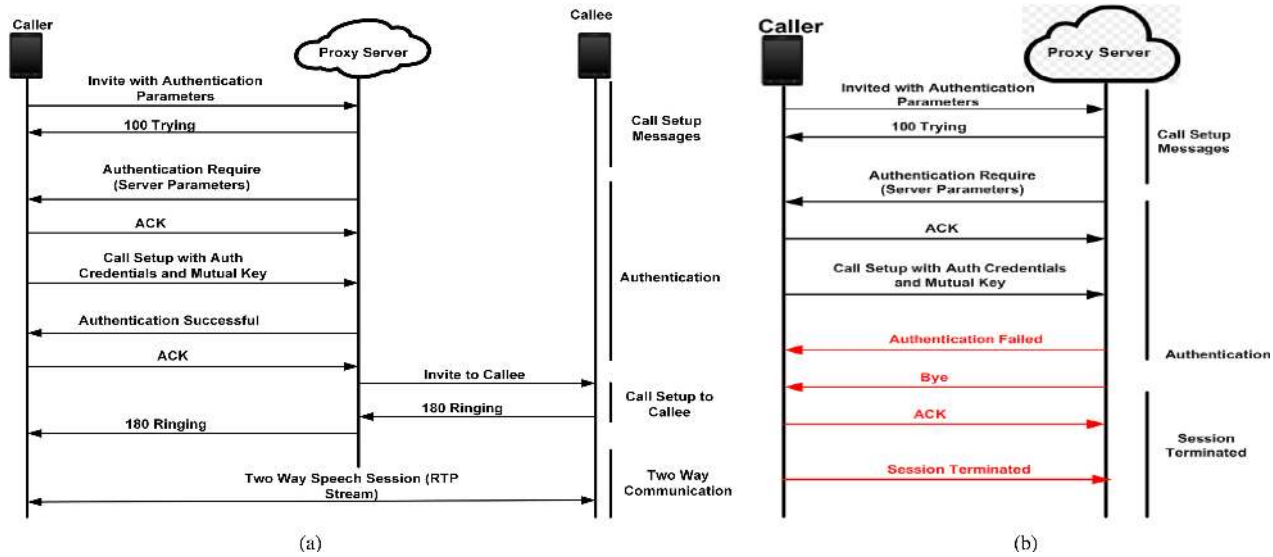

Fig. 2. Call flow sequence for the authenticated and nonauthenticated caller. (a) Successful authentication. (b) Failed authentication.

TABLE I
NOTATIONS AND ABBREVIATIONS

G	Group of prime order
g	Random generator of G
OFF-ADA	Offline active dictionary attack
OFF-PDA	Offline passive dictionary attack
FOR-SEC	Forward secrecy
ON-DA	Online dictionary attack
REP	Replay attack
D	Dictionary of passwords
p, q	Prime numbers
H	Hash function
s	The shared password between the SIP client and proxy server
$DH_g(A, B)$	The Diffie-Hellman of A and B with respect to g
Z	Zero knowledge proof of knowledge

between themselves. The unauthorized party in this process (one who controls the communication channel but does not hold the password) could not provide the successful authentication and also could not guess the password. Our scheme is based on the PAKE by Juggling protocol (or J-PAKE) [17]. Table I gives all the notations that are used in this article. The J-PAKE protocol allows two parties to establish a secure and authenticated communication based on their low-entropy shared password without requiring a PKI. The J-PAKE protocol uses the zero-knowledge proofs (ZKP) (i.e., Schnorr's signature [18]) to prove that parties are honestly following the protocol specification. J-PAKE consists of two rounds and it works as follows.

Let G denote a subgroup of $\mathbb{Z}^*_{\mathbb{P}}$ of prime order q in which the decision Diffie–Hellman problem (DDH) is intractable. Here, p and q are large primes, satisfying $q \mid p-1$. Let g be a generator in G. The parties, i.e., client and the proxy server, both agree on (G, g). Let s be their shared password, and $s \neq 0$ for any nonempty password. $Client$ selects two secret values x_1 and x_2 at random, i.e., $x_1 \in_R [1, q-1]$ and $x_2 \in_R [1, q-1]$. Similarly, proxy server selects $x_3 \in_R [1, q-1]$ and $x_4 \in_R [1, q-1]$. Note that $x_2, x_4 \neq 0$. Fig. 2 represents the flow sequence of our authentication protocol for the authenticated and nonauthenticated client. We describe working of the protocol as follows.

In the SIP authentication, the authentication process begins immediately after the caller sends the call initiation request to the proxy server. The home operator allows the client to use the network resources after the authentication is successful. We assume that the SIP client and the proxy server have agreed on the group G. We assume that the client has set a password on the system in a secure way. In this case, the client and a proxy server share a secret, i.e., a low entropy password that can be remembered by the client. The caller initiates the invite message along with the authentication credentials, i.e., [caller-ID, g^{x_1}, g^{x_2}, $Z(x_1, x_2)$]. As the proxy server receives the call request from the client, it generates the authentication required message to the client with the following information [g^{x_3}, g^{x_4}, $g^{(x_1+x_2+x_3)\cdot x_4 \cdot s}$, $Z(x_3, x_4)$, and $Z(x_4 \cdot s)$]. Upon receiving the call authentication requests, the client generates a new invite message with the authentication credentials. The authentication message from the client to the proxy server contains the following [$g^{(x_1+x_3+x_4)\cdot x_2 \cdot s}$, $Z(x_2 \cdot s)$] and $H(H(k))$, where $k = H(g^{(x_1+x_3)x_2x_4s})$. The H is a secure hash function. The proxy server upon receiving the new call setup message with the hash value and other authentication credentials would also compute its hash value as $H(H(k)$, where $k = H(g^{(x_1+x_3)x_2x_4s})$ and compare it with the received hash value. The proxy server authenticates the caller if both hash values are the same and sends back $H(k)$ as confirmation that the authentication is successful; otherwise, the proxy sends authentication failure to the client and disconnects the call request. The derived key k will serve as the mutual key between the client and the proxy server.

TABLE II
SECURITY REQUIREMENTS COMPARISON OF SCHEMES

Schemes / Security-Attacks	[16]	[17]	[18]	[19]	[11]	[13]	[12]	[20]	[21]	Digest-Auth	[22]	Ours
Resist Off-line Dictionary	No	Yes	No	Yes	Yes	Yes	No	No	No	No	Yes	Yes
Resist Sever Spoofing	No	No	No	No	Yes	No	Yes	Yes	Yes	No	Yes	Yes
Resist Replay	No	No	Yes	Yes	Yes	No	Yes	No	No	No	No	Yes
Forward Secrecy	Yes	Yes	No	Yes	No	Yes	No	Yes	No	Yes	Yes	Yes
Resist On-line Dictionary	Yes	Yes	No	Yes	Yes	Yes	No	Yes	No	Yes	Yes	Yes

C. Construction of SIP Authentication Messages

The signaling messages to perform the authentication process are shown in Fig. 2. With all of this self-enforcing authentication without PKI, the proposed scheme is compatible with the SIP RFC 3261 messages and it can be easily adaptable to any future change in the protocol by only embedding authentication parameters in the core SIP messages. The construction of SIP messages is explained as follows.

Step 1. Client → Proxy Server: The SIP client generates an SIP invite or registration message for the proxy server it directly registered with. Alice is the call initiator and Bob is the call receiver. Alice generates the invite message with the following authentication credentials.

> INVITE sip:bob@example1.com SIP/2.0
> Via: SIP/2.0/TCP client.example1.com:5060
> ;branch=z9hG4bK74b03, Max-Forwards: 70
> From Alice: sip:alice@example1.com;tag=9fxced76sl,
> **Authentication Credentials:** $g^{x_1}, g^{x_2}, Z(x_1, x_2)$
> To Bob: sip:bob@example1.com
> Call-ID: 3848276298220188511@example1.com
> CSeq: 2 INVITE
> Content-Type: application/sdp session description
> message (Continued $\cdots$)

Step 2. Proxy Server → Client: The proxy server replies client with the 407 proxy authorization required. The proxy server also presents its credentials to the client within the message. The modified authentication message is constructed as follows.

> SIP/2.0 407 Proxy Authorization Required
> Via: SIP/2.0/UDP client.example1.com:5060;
> branch=z9hG4bK74b03 ;received=192.0.2.101
> From Alice: sip:alice@example1.com;tag=9fxced76sl
> To Bob: sip:bob@example1.com;tag=876321 Call-ID:
> 2xTb9vxSit55XU7p8@example1.com CSeq: 1 IN-
> VITE
> **Proxy-Authenticate:** $g^{x_3}, g^{x_4}, g^{(x_1+x_2+x_3)\cdot x_4\cdot s}$,
> $Z(x_3, x_4), Z(x_4 \cdot s)$
> Content-Length: 0

Step 3. Client → Proxy Server: The client sends ACK message for the 407 message, together with $g^{(x_1+x_3+x_4)\cdot x_2\cdot s}$, $Z(x_2 \cdot s)$, and $H(H(k))$, where $k = H(g^{(x_1+x_3)x_2x_4s})$ and other SIP signaling related information to the proxy server.

> INVITE sip:bob@example1.com SIP/2.0
> Via: SIP/2.0/TCP client.example1.com:5060;
> branch=z9hG4bf9, Max-Forwards: 70
> From Alice: sip:alice@example1.com;tag=9fxced76sl
> To Bob: sip:bob@example1.com
> Call-ID: 3848276298220188511@example1.com
> CSeq: 2 INVITE
> **Proxy-Authorization:** $g^{(x_1+x_3+x_4)x_2\cdot s}, Z(x_2 \cdot s)$,
> $H(H(k))$
> Content-Type: application/sdp session description
> message (Continued $\cdots$)

Step 4. Proxy Server → Client: The proxy server also computes the key $k = H(g^{(x_1+x_3)x_2x_4s})$. If the hash received from the client is the same as the hash computed by the proxy server, then the client is authenticated to the proxy server, and proxy server sends the "100" ringing message to client with $H(k)$ for explicit key confirmation and the "invite" message to the callee. If the hash values of the client and proxy server are different, then the proxy server replies client with authentication failed message.

III. SECURITY ANALYSIS

In this section, we discuss the security properties of the proposed scheme. Table II presents security features of the proposed scheme along with other PAKE-based and digest authentication systems.

A. OffLine Dictionary Attack

We show that our protocol is resistant against the offline dictionary attack by both passive and active adversaries. First, we consider the scenario where Alice is honest and Bob is the active adversary trying to attack the protocol. Bob does not possess the password. He intends to gain some information about the password that would help him to perform an offline exhaustive search for the password. We show that he would not be able to accomplish this.

Let D be the dictionary and $\mathcal{B} = (\mathcal{B}_0, \mathcal{B}_1)$ be an active offline dictionary attacker against the protocol. Let K_1 be the following

probability:

$$\Pr\left[\begin{array}{c} g \xleftarrow{\$} G, X_1 \xleftarrow{\$} G, X_2 \xleftarrow{\$} G \\ s \xleftarrow{\$} D \\ (x_3, x_4, \tau) \leftarrow \mathcal{B}_0^{G,D}(g, X_1, X_2) \\ T = (\mathrm{DH}_g(X_1, X_2) * X_2^{x_3} * X_2^{x_4})^s \\ s' \leftarrow \mathcal{B}_1(T, \tau) \\ s' = s \end{array}\right]. \quad (1)$$

Note that we use $\mathrm{DH}_g(A, B)$ to denote the Diffie–Hellman of A and B with respect to g. As such, the advantage of the attacker $\mathcal{B}$ is given by $\mathrm{Adv}^G_{\mathcal{B},\mathrm{OFF-ADA1}}(\lambda) = K_1 - \frac{1}{|D|}$.

Let $\mathcal{C} = (\mathcal{C}_0, \mathcal{C}_1)$ be another offline dictionary attacker against the protocol. Let K_2 be the following probability:

$$\Pr\left[\begin{array}{c} g \xleftarrow{\$} G, X_1 \xleftarrow{\$} G, X_2 \xleftarrow{\$} G \\ s_0, s_1 \xleftarrow{\$} D \\ \text{if } s_0 = s_1 \\ \text{Abort} \\ (x_3, x_4, \tau) \leftarrow \mathcal{C}_0^{D,G}(g, X_1, X_2) \\ T_0 = (\mathrm{DH}_g(X_1, X_2) * X_2^{x_3} * X_2^{x_4})^{s_0} \\ T_1 = (\mathrm{DH}_g(X_1, X_2) * X_2^{x_3} * X_2^{x_4})^{s_1} \\ b \xleftarrow{\$} \{0, 1\} \\ b' \leftarrow \mathcal{C}_1^{D,G}(s_0, s_1, T_b, \tau) \\ b = b' \end{array}\right]. \quad (2)$$

The distinguishing advantage of $\mathcal{C}$ is given by $\mathrm{Adv}^G_{\mathcal{C},\mathrm{OFF-ADA2}}(\lambda) = K_2 - \frac{1}{2}$.

Lemma 1: $\mathrm{Adv}^G_{\mathcal{B},\mathrm{OFF-ADA1}}(\lambda) \leq 2(1 - \frac{1}{|D|}) * \mathrm{Adv}^G_{\mathcal{C},\mathrm{OFF-ADA2}}(\lambda)$.

Proof: We show that if there exists an adversary $\mathcal{B} = (\mathcal{B}_0, \mathcal{B}_1)$ against the $\mathrm{Exp}_{\mathcal{B},\mathrm{OFF-ADA1}}(\lambda)$ of (1), it could be used in the construction of another adversary $\mathcal{C}$ against the security experiment $\mathrm{Exp}^G_{\mathcal{C},\mathrm{OFF-ADA2}}(\lambda)$ of (2). $\mathcal{C}$ works as follows. It receives as input $g, X_1, X_2 \in_R G$. It invokes $\mathcal{B}_0(g, X_1, X_2)$. $\mathcal{B}_0$ outputs $x_3, x_4 \in \mathbb{Z}_p$ and the trapdoor τ. $\mathcal{C}_0$ also returns the same arguments returned by $\mathcal{B}_0$. Then, $\mathcal{C}_1$ receives as input s_0, s_1, T_b, and τ, where $T_b = (\mathrm{DH}_g(X_1, X_2) * X_2^{x_3} * X_2^{x_4})^{s_b}$. As such, $\mathcal{C}_1$ invokes $\mathcal{B}_1^D(T_b, \tau)$. $\mathcal{B}$ will return $s \in D$. If $s = s_0$, $\mathcal{C}$ returns 0, else if $s = s_1$, $\mathcal{C}$ returns 1. If $s \notin \{s_0, s_1\}$, $\mathcal{C}$ returns a random bit.

Let us now calculate the distinguishing advantage of $\mathcal{C}$. $\Pr[(\mathcal{C}_1^D() = s_b] = \Pr[\mathcal{C}_1^D(\lambda) = s_b, s = s_b) \bigcup (\mathcal{C}_1^D() = s_b, s = s_{1-b}) \bigcup (\mathcal{C}_1^D() = s_b, s \notin \{s_0, s_1\})] = \Pr[\mathcal{C}_1^D() = s_b, s = s_b)] + \Pr[\mathcal{C}_1^D() = s_b, s = s_{1-b})] + \Pr[\mathcal{C}_1^D() = s_b, s \notin \{s_0, s_1\})] = \Pr[\mathcal{C}_1^D(\lambda) = s_b | s = s_b] * \Pr[s = s_b] + \Pr[\mathcal{C}_1^D() = s_b | s = s_{1-b}] * \Pr[s = s_{1-b}] + \Pr[\mathcal{C}_1^D() = s_b | s \notin \{s_0, s_1\}] * \Pr[s \notin \{s_0, s_1\}]$. Now, $\Pr[\mathcal{C}_1^D() = s_b | s = s_b] = 1$ and $\Pr[\mathcal{C}_1^D() = s_b | s = s_{1-b}] = 0$. Also, $\Pr[\mathcal{C}_1^D() = s_b | s \notin \{s_0, s_1\}] = \frac{1}{2}$. Again, $\Pr[s = s_b] = \frac{1}{|D|} + \mathrm{Adv}^G_{\mathcal{B},\mathrm{OFF-ADA1}}(\lambda)$. $\Pr[s \notin \{s_0, s_1\}] = \Pr[s \notin \{s_b, s_{1-b}\}] = \Pr[(s \neq s_b) \bigcap (s \neq s_{1-b})] = \Pr[s \neq s_b] * \Pr[s \neq s_{1-b} | s \neq s_b] = (1 - \Pr[s = s_b]) * \Pr[s \neq s_{1-b} | s \neq s_b] = (1 - \frac{1}{|D|} - \mathrm{Adv}^G_{B,\mathrm{OFF-ADA1}}(\lambda)) * \frac{|D|-2}{|D|-1}$. Thus, $\Pr[(\mathcal{C}_1^D() = s_b] = \frac{1}{|D|} + \mathrm{Adv}^G_{\mathcal{B},\mathrm{OFF-ADA1}}(\lambda) + \frac{1}{2}((1 - \frac{1}{|D|} - \mathrm{Adv}^G_{B,\mathrm{OFF-ADA1}}(\lambda)) * \frac{|D|-2}{|D|-1}) = \frac{1}{2} + \frac{|D|}{2(|D|-1)}\mathrm{Adv}^G_{\mathcal{B},\mathrm{OFF-ADA1}}(\lambda)$. However, $\mathrm{Adv}^G_{\mathcal{C},\mathrm{OFF-ADA2}}(\lambda) \geq \Pr[(\mathcal{C}_1^D() = s_b] - \frac{1}{2} = \frac{|D|}{2(|D|-1)}\mathrm{Adv}^G_{\mathcal{B},\mathrm{OFF-ADA1}}(\lambda)$. Hence, the lemma holds. ■

Assumption 1: The DDH Assumption $\mathrm{Adv}^G_{\mathcal{A},\mathrm{DDH}}(\lambda) = M - \frac{1}{2} \leq \mathrm{negl}(\lambda)$, where M is the following probability:

$$\Pr\left[\begin{array}{c} g \xleftarrow{\$} G, A \xleftarrow{\$} G, B \xleftarrow{\$} G \\ T_0 = \mathrm{DH}_g(A, B) \\ T_1 \xleftarrow{\$} G \\ b \xleftarrow{\$} \{0, 1\} \\ b' \leftarrow \mathcal{A}(g, T_b, A, B) \\ (b = b') \end{array}\right]. \quad (3)$$

Assumption 2: $\mathrm{Adv}^G_{\mathcal{A},S\mathrm{DDH}}(\lambda) = L - \frac{1}{2}$, where L is the following probability:

$$\Pr\left[\begin{array}{c} g \xleftarrow{\$} G, A \xleftarrow{\$} G, B \xleftarrow{\$} G \\ (r, x, \tau) \leftarrow \mathcal{A}_0(g, A, B) \\ \text{if } x = 0 \vee x = 1 \\ \text{Abort} \\ \Omega_0 = \mathrm{DH}_g(A, B) * B^r \\ \Omega_1 = (\mathrm{DH}_g(A, B) * B^r)^x \\ b \xleftarrow{\$} \{0, 1\} \\ b' \leftarrow \mathcal{A}_1(\Omega_b, \tau, x) \\ b = b' \end{array}\right]. \quad (4)$$

Lemma 2: $\mathrm{Adv}^G_{\mathcal{A},S\mathrm{DDH}}(\lambda) \leq \mathrm{Adv}^G_{\mathcal{A},\mathrm{DDH}}(\lambda)$.

Proof: If $x \neq 0$, then $\mathrm{DH}_g(A, B)^x$ is a nonidentity element of G. Now according to Assumption 1, $(g, A, B, \mathrm{DH}_g(A, B) * B^r) \overset{c}{\approx} (g, A, B, R * B^r) \overset{c}{\approx} (g, A, B, R) \overset{c}{\approx} (g, A, B, R * (\mathrm{DH}_g(A, B) * B^r)^x) \overset{c}{\approx} (g, A, B, \mathrm{DH}_g(A, B) * B^r * (\mathrm{DH}_g(A, B) * B^r)^x) \overset{c}{\approx} (g, A, B, (\mathrm{DH}_g(A, B) * B^r)^{1+x})$. ■

Lemma 3: $\mathrm{Adv}^G_{\mathcal{C},\mathrm{OFF-ADA2}}(\lambda) \leq \mathrm{Adv}^G_{\mathcal{A},S\mathrm{DDH}}(\lambda)$.

Proof: We show that if there exists an adversary $\mathcal{C} = (\mathcal{C}_0, \mathcal{C}_1)$ against the security experiment $\mathrm{Exp}_{\mathcal{C},\mathrm{OFF-ADA2}}(\lambda)$, it could be used to construct another adversary $\mathcal{A}$ against the Assumption 2. $\mathcal{A}$ works as follows.

It receives as input $g, A, B \in_R \mathbb{Z}_p$. Then, it invokes $\mathcal{C}_0$ with the inputs $g, X_1 = A, X_2 = B$. $\mathcal{C}_0$ returns (x_3, x_4, τ). $\mathcal{A}_0$ computes $r = x_3 + x_4, x = s_1/s_0 - 1$, where s_1 and s_0 are randomly chosen by $\mathcal{A}_0$ from D. It returns r, x, and τ. Since, $s_0 \neq s_1$, $x \neq 0$. Now, $\mathcal{A}_1$ will receive the challenge $\Omega_b \in \{\Omega_0, \Omega_1\}$. Here, $\Omega_0 = \mathrm{DH}_g(A, B) * X_2^{x_3+x_4}$, and $\Omega_1 = (\mathrm{DH}_g(A, B) * X_2^{x_3+x_4})^{1+x}$. $\mathcal{A}$ computes $T_b = \Omega_b^{s_0}$. Note that if $b = 0$, then $\Omega_0 = \mathrm{DH}_g(A, B) * X_2^{x_3+x_4}$ and $T_b = \mathrm{DH}_g(A, B) * X_2^{x_3} * X_2^{x_4} = T_0$. Alternatively, if $b = 1$, then $\Omega_1 = (\mathrm{DH}_g(A, B) * X_2^{x_3+x_4})^{1+x}$ and $T_b = (\mathrm{DH}_g(A, B) * X_2^{x_3} * X_2^{x_4})^{s_1} = T_1$. Now, $\mathcal{A}_1$ invokes $\mathcal{C}_1(s_0, s_1, T_b, \tau)$. It will return a bit b'. $\mathcal{A}$ will return the same bit. It is easy to see that the success probability of $\mathcal{A}$ is at least that of $\mathcal{C}$. Hence, the result holds. ■

Now, we consider a passive adversary who intercepts the messages being passed between the participants and tries to infer information about the password through offline exhaustive search.

Let, the distinguishing advantage of the passive offline attacker be $\mathrm{Adv}^{G}_{\mathcal{B},\mathrm{OFF-PDA1}}(\lambda) = \Pr[\mathrm{Exp}^{G}_{\mathcal{B},\mathrm{OFF-PDA1}}(\lambda)] - \frac{1}{|D|}$, where $\Pr[\mathrm{Exp}^{G}_{\mathcal{B},\mathrm{OFF-PDA1}}(\lambda)]$ is the following probability:

$$\Pr\left[\begin{array}{c} g \xleftarrow{\$}, X_1 \xleftarrow{\$} G, X_2 \xleftarrow{\$} G, X_3 \xleftarrow{\$} G, X_4 \xleftarrow{\$} G \\ s \xleftarrow{\$} D \\ T_1 = (\mathrm{DH}_g(X_1, X_2) * \mathrm{DH}_g(X_2, X_3) * \mathrm{DH}_g(X_2, X_4))^s \\ T_2 = (\mathrm{DH}_g(X_3, X_4) * \mathrm{DH}_g(X_1, X_4) * \mathrm{DH}_g(X_2, X_4))^s \\ C = (T_1, T_2) \\ s' \leftarrow \mathcal{B}^{G,D}(C, g, X_1, X_2, X_3, X_4) \\ s = s' \end{array}\right]. \tag{5}$$

Let $\mathcal{C}$ be a passive adversary against the protocol of (6). The advantage of the adversary $\mathcal{C}$ is given by $\mathrm{Adv}^{G,D}_{\mathcal{B},\mathrm{OFF-PDA2}}(\lambda) = \Pr[\mathrm{Exp}^{G,D}_{\mathcal{B},\mathrm{OFF-PDA2}}(\lambda)] - \frac{1}{2}$, where $\Pr[\mathrm{Exp}^{G,D}_{\mathcal{B},\mathrm{OFF-PDA2}}(\lambda)]$ is the following probability:

$$\Pr\left[\begin{array}{c} g \xleftarrow{\$}, X_1 \xleftarrow{\$} G, X_2 \xleftarrow{\$} G, X_3 \xleftarrow{\$} G, X_4 \xleftarrow{\$} G \\ (s_0, s_1) \xleftarrow{\$} D \\ \text{if } s_0 = s_1 \\ \text{Abort} \\ T_1 = (\mathrm{DH}_g(X_1, X_2) * \mathrm{DH}_g(X_2, X_3) * \mathrm{DH}_g(X_2, X_4))^{s_0} \\ T_2 = (\mathrm{DH}_g(X_3, X_4) * \mathrm{DH}_g(X_1, X_4) * \mathrm{DH}_g(X_2, X_4))^{s_0} \\ T_3 = (\mathrm{DH}_g(X_1, X_2) * \mathrm{DH}_g(X_2, X_3) * \mathrm{DH}_g(X_2, X_4))^{s_1} \\ T_4 = (\mathrm{DH}_g(X_3, X_4) * \mathrm{DH}_g(X_1, X_4) * \mathrm{DH}_g(X_2, X_4))^{s_1} \\ C_0 = (T_1, T_2), C_1 = (T_3, T_4) \\ b \xleftarrow{\$} \{0, 1\} \\ b' \leftarrow \mathcal{C}^{G,D}(C_b, s_0, s_1, g, X_1, X_2, X_3, X_4) \\ b = b' \end{array}\right]. \tag{6}$$

Lemma 4: $\mathrm{Adv}^{G}_{\mathcal{B},\mathrm{OFF-PDA1}}(\lambda) \leq 2(1 - \frac{1}{|D|})\mathrm{Adv}^{G}_{\mathcal{C},\mathrm{OFF-PDA2}}(\lambda)$.

Proof: The proof is same as the proof of Lemma 1. ■

B. Online Dictionary Attack

In this section, we show that our scheme is secure against an online dictionary attack. Consider the following security experiment:

$Exp^{G,D}_{\mathcal{B},ON-DA}(\lambda)$

$g, X_1, X_2, \xleftarrow{\$} G$
$(x_3, x_4, \tau) = \mathcal{B}^{G,D}_0(g, X_1, X_2)$
$s \xleftarrow{\$} D$
$T = (DH_g(X_1, X_2) * X_2^{x_3} * X_2^{x_4})^s$
$L = (DH_g(X_1, X_2) * X_2^{x_3})^{x_4 s}$
$L' = \mathcal{B}^{G,D}_1(T, \tau)$
Return $L = L'$

The advantage of the adversary $\mathcal{B}$ in computing the secret key is given by $\mathrm{Adv}^{G,D}_{\mathcal{B},\mathrm{ON-DA}}(\lambda) = \Pr[\mathrm{Exp}^{G,D}_{\mathcal{B},\mathrm{ON-DA}}(\lambda)] - \frac{1}{|D|}$.

Lemma 5: $\mathrm{Adv}^{G,D}_{\mathcal{B},\mathrm{ON-DA}}(\lambda) \leq \mathrm{Adv}^{G,D}_{\mathcal{A},\mathrm{OFF-ADA1}}(\lambda)$.

Proof: We show that if there exists an online dictionary attacker $\mathcal{B} = (\mathcal{B}_0, \mathcal{B}_1)$, then it could be used to construct an adversary $\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1)$ against the security experiment $\mathrm{Exp}^{G}_{\mathcal{A},\mathrm{OFF-ADA1}}(\lambda)$. $\mathcal{A}$ works as follows. When $\mathcal{A}_0$ receives g, X_1, X_2, it invokes $\mathcal{B}^{G,D}_0(g, X_1, X_2)$. It returns $x_3, x_4, \tau' = \tau \bigcup \{x_3, x_4\}$. $\mathcal{A}_0$ returns the same arguments. Then, $\mathcal{A}_1$ receives as input T, τ', where $T = (\mathrm{DH}_g(X_1, X_2) * X_2^{x_3} * X_2^{x_4})^s$ for some $s \in D$. $\mathcal{A}_1$ invokes $\mathcal{B}^{G,D}_1(T, \tau)$. $\mathcal{B}_1$ will return $L = (\mathrm{DH}_g(X_1, X_2) * X_2^{x_3})^{x_4 s}$. Now, $\mathcal{A}_1$ computes $X_2^s = (T^{x_4}/L)^{1/x_4^2}$. Now, $\mathcal{A}$ can find s using brute force search over all the elements in D. This search will be feasible since $|D| \in \mathrm{poly}(\lambda)$. Now, $\mathcal{A}_1$ can output s. $\Pr[\mathrm{Exp}^{G,D}_{\mathcal{A},\mathrm{OFF-ADA1}}(\lambda) = 1] = \Pr[\mathcal{A}_0(T, \tau') = s] \geq \Pr[\mathcal{B}_1(T, \tau) = L] = \Pr[\mathrm{Exp}^{G,D}_{\mathcal{B},\mathrm{ON-DA}}(\lambda) = 1]$. Hence, $\mathrm{Adv}^{G,D}_{\mathcal{A},\mathrm{OFF-ADA1}}(\lambda) \geq \mathrm{Adv}^{G,D}_{\mathcal{B},\mathrm{ON-DA}}(\lambda)$. ■

Thus, the attacker would not be able to establish the correct secret key if it chooses a wrong password.

C. Forward Secrecy

In this section, we show that our scheme provides forward secrecy. Hence, if an attacker gets to learn the shared password between the two parties, she will be able to compromise the secret keys of previous sessions with negligible probability. Let $\mathcal{B}$ be an attacker against the forward secrecy property of our scheme. As such, the advantage of the adversary to compromised a previously computed shared key is given by $\mathrm{Adv}^{G,D}_{\mathcal{B},\text{FOR-SEC}}(\lambda) = \Pr[\mathrm{Exp}^{G,D}_{\mathcal{B},\text{FOR-SEC}}(\lambda) = 1]$. Our scheme is forward-secure if the following holds:

$$\mathrm{Adv}^{G,D}_{\mathcal{B},\text{FOR-SEC}}(\lambda) \leq \mathrm{negl}(\lambda).$$

$Exp^{G,D}_{\mathcal{B},FOR-SEC}(\lambda)$

$g \xleftarrow{\$}, X_1 \xleftarrow{\$} G, X_2 \xleftarrow{\$} G, X_3 \xleftarrow{\$} G, X_4 \xleftarrow{\$} G$
$s \xleftarrow{\$} D$
$L_1 = (DH_g(X_1, X_2) * DH_g(X_2, X_3) * DH_g(X_2, X_4))^s$
$L_2 = (DH_g(X_3, X_4) * DH_g(X_1, X_4) * DH_g(X_2, X_4))^s$
$B = DH_g(X_1 * X_3, X_2, X_4)^s$
$B' = \mathcal{B}^{G,D}(g, X_1, X_2, X_3, X_4, L_1, L_2, s)$
Return B=B'

Assumption 3:

$Exp^{G}_{\mathcal{B},CDH}(\lambda)$

$g \xleftarrow{\$}, A \xleftarrow{\$} G, B \xleftarrow{\$} G, C \xleftarrow{\$} G$
$E_1 = DH_g(A, B), E_2 = DH_g(B, C), E_3 = DH_g(A, C)$
$T = DH_g(A, B, C)$
$T' = \mathcal{B}^{G}(g, A, B, C, E_1, E_2, E_3)$
Return T=T'

According to the Computation Diffie–Hellman assumption, for all PPT adversary $\mathcal{A}$, $\mathrm{Adv}^{G}_{\mathcal{A},\mathrm{CDH}}(\lambda) = \Pr[\mathrm{Exp}^{G}_{\mathcal{A},\mathrm{CDH}}(\lambda) = 1] \leq \mathrm{negl}(\lambda)$.

Lemma 6: $\mathrm{Adv}^{G,D}_{\mathcal{B},\text{FOR-SEC}}(\lambda) \leq \mathrm{Adv}^{G}_{\mathcal{A},\mathrm{CDH}}(\lambda)$.

Proof: We show that if there exists a probabilistic polynomial-time (PPT) adversary $\mathcal{B}$ against the security experiment $\mathrm{Exp}^{G}_{\mathcal{B},\text{FOR-SEC}}(\lambda)$, it could be used to construct

another PPT adversary $\mathcal{A}$ against the security experiment $\text{Exp}^{G}_{\mathcal{A},\text{CDH}}(\lambda)$. $\mathcal{A}$ works as follows. It receives as inputs $g, A, B, C \in_R G$ and E_1, E_2, E_3. It selects $a \in_R \mathbb{Z}_p$ and computes $X_1 = g^a$. It sets $X_2 = B, X_4 = C$, and $X_3 = A/X_1$. It also selects random $s \in D$, and computes $(L_1 = E_1 * E_2)^s = (\text{DH}_g(A, B) * \text{DH}_g(B, C))^s$, and $L_2 = (E_3 * E_2)^s = (\text{DH}_g(A, C) * \text{DH}_g(B, C))^s$. Now, $\mathcal{A}$ invokes $\mathcal{B}^{G,D}(g, X_1, X_2, X_3, X_4, L_1, L_2, s)$. $\mathcal{B}$ will return $B' = \text{DH}_g(X_1 * X_3, X_2, X_4)^s = \text{DH}_g(A, B, C)^s$. $\mathcal{A}$ can compute $\text{DH}_g(A, B, C) = (B')^{1/s}$. Thus, $\text{Adv}^{G}_{\mathcal{A},\text{CDH}}(\lambda) \geq \text{Adv}^{G,D}_{\mathcal{B},\text{FOR-SEC}}(\lambda)$. ■

D. Replay Attack

In replay attack, the adversary can use an older key and he can then replay the messages. Let $\mathcal{B}$ be an adversary who launches the replay attack on our scheme. Her intention is to obtain the secret password shared by Alice and Bob. The advantage of $\mathcal{B}$ in obtaining the password is given by $\text{Adv}^{G,D}_{\mathcal{B},\text{REP1}}(\lambda) = \Pr[\text{Exp}^{G,D}_{\mathcal{B},\text{REP1}}(\lambda) = 1] - \frac{1}{|D|}$.

$Exp^{G,D}_{\mathcal{B},REP1}(\lambda)$

$g, X_1, X_2, X_1', X_2', X_3, X_4 \xleftarrow{\$} G$
$s \xleftarrow{\$} D$
$T_1 = (DH_g(X_1, X_2) * DH_g(X_2, X_3) * DH_g(X_2, X_4))^s$
$T_2 = (DH_g(X_3, X_4) * DH_g(X_1, X_4) * DH_g(X_2, X_4))^s$
$T_3 = (DH_g(X_1', X_2') * DH_g(X_2', X_3) * DH_g(X_2', X_4))^s$
$s' = \mathcal{B}^{G,D}(g, X_1, X_2, X_1', X_2', X_3, X_4, T_1, T_2, T_3)$
Return $s = s'$

Lemma 7: $\text{Adv}^{G,D}_{\mathcal{B},\text{REP1}}(\lambda) \leq \text{Adv}^{G,D}_{\mathcal{A},\text{OFF-PDA1}}(\lambda)$.

Proof: We show that if there exists an adversary $\mathcal{B}$ against the security experiment $\text{Exp}^{G,D}_{\mathcal{B},\text{REP1}}(\lambda)$, then it could be used in the construction of $\mathcal{A}$, an adversary against the security experiment $\text{Exp}^{G,D}_{\mathcal{A},\text{OFF-PDA1}}(\lambda)$. $\mathcal{A}$ works as follows. It receives as inputs g, X_1, X_2, X_3, X_4, and a challenge $C = (T_1, T_2)$, where

$$T_1 = (\text{DH}_g(X_1, X_2) * \text{DH}_g(X_2, X_3) * \text{DH}_g(X_2, X_4))^s$$
$$T_2 = (\text{DH}_g(X_3, X_4) * \text{DH}_g(X_1, X_4) * \text{DH}_g(X_2, X_4))^s .$$

$\mathcal{A}$ computes $X_1' = X_1^a (X_3 * X_4)^{a-1}$, and $X_2' = X_2^b$ for some random $a, b \in_R \mathbb{Z}_p$. Now, $\mathcal{A}$ sets $T_3 = T_1^{ab}$ and $C' = (T_1, T_2, T_3)$. Then, $\mathcal{A}$ invokes $\mathcal{B}^{G,D}(C', g, X_1, X_2, X_1', X_2', X_3, X_4)$ and returns what $\mathcal{B}$ returns. It is easy to see that $\text{Adv}^{G,D}_{\mathcal{B},\text{REP1}}(\lambda) \leq \text{Adv}^{G,D}_{\mathcal{A},\text{OFF-PDA1}}(\lambda)$. ■

Now, we consider another replay attacker whose wish is to establish a secret key with Alice. The adversary intercepts the messages between Alice and Bob in a particular session. Then, she uses those messages to launch replay attack with the intention to establish a shared key with Alice. We consider the following security experiment $\text{Exp}^{G,D}_{\mathcal{B},\text{REP2}}(\lambda)$. The advantage of the adversary $\mathcal{B}$ in being able to establish a secret key is given by $\text{Adv}^{G,D}_{\mathcal{B},\text{REP2}}(\lambda) = \Pr[\text{Exp}^{G,D}_{\mathcal{B},\text{REP2}}(\lambda) = 1]$.

Lemma 8: $\text{Adv}^{G,D}_{\mathcal{B},\text{REP2}}(\lambda) \leq \text{Adv}^{G,D}_{\mathcal{A},\text{FOR-SEC}}(\lambda)$.

Proof: We show that if there exists an adversary $\mathcal{B}$ against the security experiment $\text{Exp}^{G,D}_{\mathcal{B},\text{REP2}}(\lambda)$, it could be used in the construction of another adversary $\mathcal{A}$ against the security experiment $\text{Exp}^{G,D}_{\mathcal{A},\text{FOR-SEC}}(\lambda)$. $\mathcal{A}$ receives as inputs $g, X_1, X_2, X_3, X_4, L_1, L_2, s$, where

$$L_1 = (\text{DH}_g(X_1, X_2) * \text{DH}_g(X_2, X_3) * \text{DH}_g(X_2, X_4))^s$$
$$L_2 = (\text{DH}_g(X_3, X_4) * \text{DH}_g(X_1, X_4) * \text{DH}_g(X_2, X_4))^s$$

$Exp^{G,D}_{\mathcal{B},REP2}(\lambda)$

$g, X_1, X_2, X_1', X_2', X_3, X_4 \xleftarrow{\$} G$
$s \xleftarrow{\$} D$
$L_1 = (DH_g(X_1, X_2) * DH_g(X_2, X_3) * DH_g(X_2, X_4))^s$
$L_2 = (DH_g(X_3, X_4) * DH_g(X_1, X_4) * DH_g(X_2, X_4))^s$
$L_3 = (DH_g(X_1', X_2') * DH_g(X_2', X_3) * DH_g(X_2', X_4))^s$
$B = DH_g(X_1' * X_3, X_2', X_4)^s$
$B' = \mathcal{B}^{G,D}(g, X_1, X_2, X_3, X_4, X_1', X_2', L_1, L_2, L_3)$
Return B=B'

$\mathcal{A}$ selects $X_1' = X_1$, and $X_2' = X_2^b$ for some random $b \in_R \mathbb{Z}_p$. It computes $L_3 = (L_1)^b$. Then, it invokes $\mathcal{B}^{G,D}(g, X_1, X_2, X_3, X_4, X_1', X_2', L_1, L_2, L_3)$. $\mathcal{B}$ will return $B = \text{DH}_g(X_1' * X_3, X_2', X_4)^s = (\text{DH}_g(X_1 * X_3, X_2, X_4)^s)^b$. $\mathcal{A}$ will return $B^{1/b}$. It is easy to see that the success probability of $\mathcal{A}$ is at least that of $\mathcal{B}$. Hence, the lemma holds. ■

IV. Computation and Bandwidth Overheads

In this section, we analyze the computation and bandwidth overheads of the proposed scheme for its cryptographic operations. The client needs to perform around 14 exponentiation during the authentication process. Four exponentiation for g^{x_1}, g^{x_2}, and $Z(x_1, x_2)$, four exponentiation to prove the ZKPs of x_3 and x_4 from the server, two exponentiation to verify the ZKP of $x_4 \cdot s$, two exponentiation for computing $g^{(x_1+x_3+x_4)\cdot(x_2\cdot s)}$ and the ZKP for $x_2 \cdot s$, and two exponentiation to compute the value of final key k. The proxy server also performs 14 exponentiation to prove the variables from the client, generating the authentication credentials, and mutual key k. We computed time for generating the authentication parameters with the single-core of Intel i-7 CPU (3.4 GHz) system, having 8-GB memory on a Windows 10 operating system. We implemented the protocol in the Java using NIST curve P-256 and bouncy-castle elliptical curve library for the cryptography. The client and server take around 30 ms to generate the authentication credentials in the first round, and 25 ms in the second round.

In terms of bandwidth, the client and the proxy server exchanged information to each other in two rounds. In the first invite message, the client exchanges g^{x_1}, g^{x_2} the $Z(x_1, z_2)$ to the proxy server. This exchange requires around 692 bytes. The proxy server initiates authentication required message with g^{x_3}, g^{x_4}, $g^{(x_1+x_2+x_3)\cdot x_4\cdot s}$, $Z(x_4 \cdot s)$, and $Z(x_3, z_4)$ to the client. This exchange requires 1 kb. Finally, the client sends $g^{(x_1+x_3+x_4)\cdot x_2\cdot s}$, $Z(x_2 \cdot s)$ that requires around 350 bytes of data. In summary, the client requires to exchange around 1 kb of data to proxy and receive 1 kb of data from the proxy server for the authentication.

V. Related Work

The simplest method to achieve the authentication in the SIP-based VoIP or NGN is to utilize the challenge-response

mechanism [Internet Engineering Task Force (IETF) RFC 2617] [26]. In this mechanism, the SIP call processing engine or the proxy server on receiving the call request message from the SIP user initiates the challenge to the user to prove his identity. The client responds to the proxy server with authentication messages. This authentication mechanism has some security problems: for instances, it is vulnerable to offline password guessing attack, server spoofing, falsifying the identity of the server to obtain the secret information of user, etc. Table II presents a comparison of our scheme with other proposed systems for a number of security requirements. It can be seen from Table II that digest-based schemes are vulnerable to different types of security attacks, i.e., offline password guessing attacks, server spoofing, replay attack, etc. It can also be seen that many of the proposed schemes only provide resistance against a few features. However, the proposed scheme not only provides resistance against the listed attacks but also incurs substantially small overheads.

Several public-key cryptography-based systems have also been proposed to ensure secure authentication. Chou-Chen *et al.* [14] proposed an authentication scheme based on the Diffie–Hellman key change mechanism [27]. However, the scheme is vulnerable to an offline password-guessing attack and stolen verifier attack. [15], [21], [28]. Furthermore, Yang *et al.'s* scheme requires computational resources at the client and server. Liufei *et al.* [15] adopted elliptical curve cryptography (ECC) to facilitate the authentication and key agreement between the SIP client and the proxy server. The mechanism provides mutual authentication and provable security but is vulnerable to the offline password guessing attack because the session key is not used in the authentication responses [29]. Yi-Pin *et al.* [13] proposed the authentication scheme based on self-certified public keys on elliptic curves. The scheme does not require PKI for the cryptographic keys and parameters. However, the scheme requires the smart card to stores the parameters. Srinivasan *et al.* [10] use PKI and a strong one-way hash function to authenticate the client in the SIP network. However, the scheme is vulnerable to the stolen verifier attack. Liping *et al.* [23] proposed a flexible password-authenticated key agreement for the session initiation employing a smart card. The smart card holds all the information related to cryptographic parameters. However, the scheme is vulnerable to the impersonation attack. Qi *et al.* [24] improved scheme of Liping *et al.* and supported defense against the impersonation attack. Ni *et al.* [19] proposed signature-based authentication and key agreement scheme for SIP-based networks. The public keys are generated through the identity of the client and the proxy server.

Jia *et al.* [30] use random nonces for authenticating the SIP client with the server. However, the scheme is vulnerable to the Denning–Sacco attack, the stolen-verifier attack, and the offline password guessing attack. Eun-Jun and Kee-Young improved the basic scheme of Aytunc and Ibrahim [31] by using the random number for the public key, which is not happening in the Aytunc and Ibrahim scheme. Tien-ho *et al.* [12] proposed an ECC-based authentication mechanism that protects the user from the server spoofing attack and session hijacking attack. The scheme is based on using a smart card to minimize the computation load, however, it is vulnerable to password guessing attack. Eun-Jun and Kee-Young adopted an elliptic curve discrete logarithm problem to address the problem of offline password guessing attacks, Denning–Sacco attack, and stolen-verifier attacks of SIP authentication. Zhang *et al.* [32] proposed an authentication scheme based on the elliptic curve with the inherent property of anonymity for the SIP client. However, the scheme does not support mutual authentication and is vulnerable to insider attack [22]. Recently, Shuming *et al.* [33] proposed the scheme on the top of Zhang *et al.* [32] that provide resistance against offline password guessing and insider attacks. Hsiu-Lien [34] proposed a scheme that uses a smart card along with elliptic curve cryptography for the SIP authentication. However, the scheme is vulnerable to the offline password guessing attack, user impersonation attack, and server impersonation attack [35]. Hang *et al.* [20] proposed modifications in [32] to overcome the issue of a server spoofing attack. Chaudhry *et al.* [36] proposed the privacy-preserving version for [32] and [35] based on the elliptic curve cryptography.

The successful authentication can also solve the problem of identity spoofing that causes the loss of millions every year. Cybercriminals can easily modify identity and pretend to be a legitimate entity to fool the user at the other end. Typically, with the spoofed identity, criminals fool people into thinking that they are interacting with the legitimate entity, e.g., their bank, or the police. Currently, the IETF is favoring a PKI-based approach to solve the caller ID spoofing problem. In 2018, it published a new technical standard [37] that defines a telephone certificate based on X.509. This is regarded as the first step toward a full PKI deployment in telephony systems. A certificate authority-based solution is proposed in [38] where the originating operators present the certificate of ownership through the call routing mechanism. Bradley *et al.* [39] propose to adopt SSL/TLS for the caller ID authentication. The schemes assume a trusted server, with which the caller can register its identity through an SSL/TLS connection. In general, solutions in this category require a PKI to bind the caller ID with a telephone using a public key certificate.

VI. Conclusion

In this article, we proposed a new authentication scheme for authenticating clients/end users in the SIP-based NGNs. The proposed scheme enables the proxy server and the SIP clients to exchange the authentication messages over an open and insecure network. We adopted the password-based authentication mechanism along with ZKPs to perform the authentication process. The scheme does not require PKI or the smart card for the cryptographic parameters and has inherent properties of self-enforcement. The proposed authentication scheme provides effective security against different types of attacks and does not incur substantial computational overheads. The scheme can also provide a way for the parameters to be used for the end-to-end encryption of speech content between the communicating parties. As part of the future work, we are developing a working prototype that involves the real SIP server and the SIP clients.

References

[1] C. Perera, M. Barhamgi, S. De, T. Baarslag, M. Vecchio, and K. R. Choo, "Designing the sensing as a service ecosystem for the Internet of Things," *IEEE Internet Things Mag.*, vol. 1, no. 2, pp. 18–23, Dec. 2018.

[2] M. Barhamgi, C. Perera, C. Ghedira, and D. Benslimane, "User-centric privacy engineering for the internet of things," *IEEE Cloud Comput.*, vol. 5, no. 5, pp. 47–57, Sep. 2018.

[3] J. Rosenberg *et al.*, "SIP: Session initiation protocol," IETF RFC 3261, United States, 2002.

[4] S. Ehlert, D. Geneiatakis, and T. Magedanz, "Survey of network security systems to counter SIP-based denial-of-service attacks," *Comput. Secur.*, vol. 29, no. 2, pp. 225–243, 2010.

[5] M. A. Azad and R. Morla, "Caller-REP: Detecting unwanted calls with caller social strength," *Comput. Secur.*, vol. 39, pp. 219–236, Nov. 2013. [Online]. Available: http://dx.doi.org/10.1016/j.cose.2013.07.006

[6] M. Ajmal, S. Bag, S. Tabassum, and F. Hao, "privy: Privacy preserving collaboration across multiple service providers to combat telecoms spam," *IEEE Trans. Emerg. Topics Comput.*, to be published, doi: 10.1109/TETC.2017.2771251.

[7] S. Salsano, L. Veltri, and D. Papalilo, "SIP security issues: The SIP authentication procedure and its processing load," *IEEE Netw.*, vol. 16, no. 6, pp. 38–44, Nov. 2002.

[8] I. Dacosta and P. Traynor, "Proxychain: Developing a robust and efficient authentication infrastructure for carrier-scale VOIP networks," in *Proc. USENIX Conf.*, 2010, pp. 10–10.

[9] T. Maitra, D. Giri, and R. N. Mohapatra, "SAS-SIP: A secure authentication scheme based on ECC and a fuzzy extractor for session initiation protocol," *Cryptologia*, vol. 43, no. 3, pp. 212–232, 2019. [Online]. Available: https://doi.org/10.1080/01611194.2018.1548391

[10] R. Srinivasan, V. Vaidehi, K. Harish, K. LakshmiNarasimhan, S. LokeshwerBabu, and V. Srikanth, "Authentication of signaling in VIOP applications," in *Proc. Asia-Pacific Conf. Commun.*, Oct. 2005, pp. 530–533.

[11] A. Mohammadi-nodooshan, Y. Darmani, R. Jalili, and M. Nourani, "A robust and efficient SIP authentication scheme," in *Advances in Computer Science and Engineering*. Berlin, Germany: Springer, 2009, pp. 551–558.

[12] T.-H. Chen, H.-L. Yeh, P.-C. Liu, H.-C. Hsiang, and W.-K. Shih, "A secured authentication protocol for SIP using elliptic curves cryptography," in *Communication and Networking*, T.-H. Kim, A.C.-C. Chang, M. Li, C. Rong, C. Z. Patrikakis, and D. Ślęzak, Eds., Berlin, Germany: Springer, 2010, pp. 46–55.

[13] Y.-P. Liao and S.-S. Wang, "A new secure password authenticated key agreement scheme for SIP using self-certified public keys on elliptic curves," *Comput. Commun.*, vol. 33, no. 3, pp. 372–380, 2010.

[14] C.-C. Yang, R.-C. Wang, and W.-T. Liu, "Secure authentication scheme for session initiation protocol," *Comput. Secur.*, vol. 24, no. 5, pp. 381–386, 2005.

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

[16] E. Yoon and K. Yoo, "Cryptanalysis of DS-SIP authentication scheme using ECDH," in *Proc. Int. Conf. New Trends Inf. Service Sci.*, Jun. 2009, pp. 642–647.

[17] F. Hao and P. Ryan, *J-PAKE: Authenticated Key Exchange Without PKI*. Berlin, Germany: Springer, 2010, pp. 192–206.

[18] C. P. Schnorr, "Efficient signature generation by smart cards," *J. Cryptol.*, vol. 4, no. 3, pp. 161–174, Jan. 1991.

[19] L. Ni, G. Chen, and J. Li, "A pairing-free identity-based authenticated key agreement mechanism for SIP," in *Proc. Int. Conf. Netw. Comput. Inf. Secur.*, vol. 1, May 2011, pp. 209–217.

[20] H. Tu, N. Kumar, N. Chilamkurti, and S. Rho, "An improved authentication protocol for session initiation protocol using smart card," *Peer-to-Peer Netw. Appl.*, vol. 8, no. 5, pp. 903–910, Sep. 2015.

[21] E. Yoon and K. Yoo, "A new authentication scheme for session initiation protocol," in *Proc. Int. Conf. Complex, Intell. Softw. Intensive Syst.*, Mar. 2009, pp. 549–554.

[22] Y. Lu, L. Li, H. Peng, and Y. Yang, "A secure and efficient mutual authentication scheme for session initiation protocol," *Peer-to-Peer Netw. Appl.*, vol. 9, no. 2, pp. 449–459, Mar. 2016. [Online]. Available: https://doi.org/10.1007/s12083-015-0363-x

[23] L. Zhang, S. Tang, and Z. Cai, "Efficient and flexible password authenticated key agreement for voice over internet protocol session initiation protocol using smart card," *Int. J. Commun. Syst.*, vol. 27, no. 11, pp. 2691–2702, Nov. 2014.

[24] 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." *Int. J. Commun. Syst.*, vol. 28, no. 7, pp. 1340–1351, May 2015.

[25] L. Zhang, S. Tang, and S. Zhu, "An energy efficient authenticated key agreement protocol for SIP-based green VOIP networks," *J. Netw. Comput. Appl.*, vol. 59, pp. 126–133, Jan. 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804515001666

[26] P. J. Franks *et al.*, "HTTP authentication: Basic and digest access authentication," RFC 2617, Jun. 1999. [Online]. Available: https://rfc-editor.org/rfc/rfc2617.txt

[27] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Trans. Inf. Theory*, vol. 22, no. 6, pp. 644–654, Sep. 2006.

[28] H. Jo, Y. Lee, M. Kim, S. Kim, and D. Won, "Off-line password-guessing attack to Yang's and Huang's authentication schemes for session initiation protocol," in *Proc. 5th Int. Joint Conf. INC, IMS IDC*, Aug. 2009, pp. 618–621.

[29] H. H. Kilinc and T. Yanik, "A survey of SIP authentication and key agreement schemes," *IEEE Commun. Surv. Tuts.*, vol. 16, no. 2, pp. 1005–1023, 2Q 2014.

[30] J. L. Tsai, "Efficient Nonce-based authentication scheme for session initiation protocol," *Int. J. Netw. Secur.*, vol. 9, pp. 12–16, 2009.

[31] A. Durlanik and I. Sogukpinar, "SIP authentication scheme using ECDH," *Int. J. Comput., Elect., Autom., Control Inf. Eng.*, vol. 1, no. 8, pp. 2672–2675, 2007.

[32] Z. Zhang, Q. Qi, N. Kumar, N. Chilamkurti, and H.-Y. Jeong, "A secure authentication scheme with anonymity for session initiation protocol using elliptic curve cryptography," *Multimedia Tools Appl.*, vol. 74, no. 10, pp. 3477–3488, May 2015.

[33] S. Qiu, G. Xu, Y. Guo, and M. Zhang, "Cryptanalysis and improvement of 2 mutual authentication schemes for session initiation protocol," *Int. J. Commun. Syst.*, vol. 31, no. 10, 2018, Art. no. e3568.

[34] H.-L. Yeh, T.-H. Chen, and W.-K. Shih, "Robust smart card secured authentication scheme on SIP using elliptic curve cryptography," *Comput. Standards Interfaces*, vol. 36, no. 2, pp. 397–402, 2014.

[35] M. S. Farash, S. Kumari, and M. Bakhtiari, "Cryptanalysis and improvement of a robust smart card secured authentication scheme on SIP using elliptic curve cryptography," *Multimedia Tools Appl.*, vol. 75, no. 8, pp. 4485–4504, Apr. 2016.

[36] S. A. Chaudhry, H. Naqvi, M. Sher, M. S. Farash, and M. U. Hassan, "An improved and provably secure privacy preserving authentication protocol for SIP," *Peer-to-Peer Netw. Appl.*, vol. 10, no. 1, pp. 1–15, Jan. 2017.

[37] J. Peterson and S. Turner, "Secure telephone identity credentials: Certificates," Internet Engineering Task Force (IETF), Fremont, CA, USA, RFC 8226, Feb. 2018.

[38] H. Tu, A. Doupé, Z. Zhao, and G. Ahn, "Toward authenticated caller ID transmission: The need for a standardized authentication scheme in q.731.3 calling line identification presentation," in *Proc. ITU Kaleidoscope, ICTs Sustain. World*, 2016, pp. 1–8.

[39] B. Reaves, L. Blue, and P. Traynor, "AuthLoop: End-to-end cryptographic authentication for telephony over voice channels," in *Proc. 25th USENIX Secur. Symp.*, Austin, TX, USA, 2016, pp. 963–978.

Muhammad Ajmal Azad received the Ph.D. degree in electrical and computer engineering from the University of Porto, Porto, Portugal, in 2016.

He is a Lecturer in Cybersecurity with the Department of Computer Science and Mathematics, University of Derby, Derby, U.K. His research interests include data-driven network security, identity linking, network data analysis, and applying machine learning for detecting cybercrimes and spams.

Samiran Bag received the M.Tech. and Ph.D. degrees in computer science from Indian Statistical Institute, New Delhi, India, in 2015.

He is currently a Research Fellow with the Department of Computing Science, University of Warwick, Coventry, U.K. His main research interests include electronic voting, cryptocurrency, and secure multiparty computation.

Charith Perera received the Ph.D. degree in computer science from Australian National University, Canberra, Australia, in 2014.

He is currently a Lecturer of Computer Science and Informatics with the School of Computer Science and Informatics, Cardiff University, Cardiff, U.K. His research interests include security and privacy in the Internet of Things.

Mahmoud Barhamgi received the Ph.D. degree in information and communication technology from Claude Bernard University Lyon 1, Villeurbanne, France, in 2010.

He is currently an Associate Professor in Computer Science with Claude Bernard University Lyon 1. His research interests include privacy preservation in service-oriented architecture, web, and cloud environments.

Feng Hao received the Ph.D. degree in computer science from University of Cambridge, Cambridge, U.K., in 2007.

He is currently a Professor in Security Engineering with the Department of Computing Science, The University of Warwick, Coventry, U.K. He worked in security industry for several years before joining the Faculty, Newcastle University, Newcastle upon Tyne, U.K., as a Lecturer in 2010, and became a Reader in 2014 and a Professor in 2018. His research interests include applied cryptography, security engineering, and efficient computing.

Prof. Hao has been serving as an Associate Editor for *IEEE Security and Privacy*, since 2013.