Secure and Efficient Authentication Scheme for VoIP Communications

Osama S. Younes[1,2]
[1] Computer Engineering Department, Faculty of Computer and Information Technology
University of Tabuk
Tabuk, SA
[2] Information Technology Department, Faculty of Computers and Information
Menoufia University,
Menoufia, Egypt
o_younes@ut.edu.sa

***Abstract*— Over the past decade, Voice over IP (VoIP) has evolved from being a voice communication system into a robust unified communications engine. All VoIP devices rely on a single protocol known as the Session Initiation Protocol (SIP). SIP defines the protocols and communication methods used to establish phone calls and is commonly recognized as an IP-based multimedia communication protocol. Among the various security services recommended for SIP, authentication stands as the most essential. While numerous schemes have been introduced in the literature to enhance SIP security, many of them exhibit high computational costs, which renders them less scalable. Additionally, these schemes often lack compatibility with standard SIP protocols. In this study, we propose a novel authentication and key agreement scheme aimed at securing SIP communications. This proposed scheme is based on the Kerberos protocol and is called Kerberos-SIP (K-SIP). Our scheme significantly reduces computational costs, introduces a single sign-on capability, enables two-way authentication, facilitates secure key agreements among parties, maintains compatibility with SIP, and mitigates various SIP attacks. We thoroughly analyze the security properties inherent in the proposed scheme and, in tandem, investigate its performance characteristics.**

Keywords—VoIP, SIP, Kerberos, SIP authentication, SIP Security, VoIP security

I. INTRODUCTION

In contemporary times, VoIP (Voice over IP) is a technology used to facilitate real-time communications, such as voice calls and video calls. VoIP enables voice communications to be transmitted over an IP network. To start, manage, and end multimedia sessions between participants in a VoIP service, a session initiation (SIP) protocol is used. SIP is a text-based client/server signalling protocol [1]. not only does VoIP reduce telecommunications costs, but it also provides benefits to businesses that are not available through traditional telephone systems.

SIP has found application in numerous contexts, including file transfers, video conferences, voice/video distribution, and online gaming [2]. Given SIP's compatibility with video telephony, Internet of Things (IoT) imaging devices can be configured to operate as SIP endpoints. This configuration allows users to access the video feed from such devices on a VoIP telephone or a mobile phone.

SIP plays a critical role in various VoIP communications scenarios. However, the inherent vulnerability of SIP makes it susceptible to hacking attempts, underscoring the importance of ensuring robust VoIP security to safeguard the overall system. As the utilization of VoIP continues to expand, the issue of SIP security has become increasingly significant [3-4]. Authentication emerges as a pivotal aspect of fortifying SIP against threats. When employing SIP, a client initiates a request to the server for setting up a voice call session, relying on shared or widely recognized parameters. It's crucial for the client to ascertain that it's establishing a connection with the legitimate SIP user agent or server, rather than falling prey to an attacker [5-7].

Creating a secure authentication and key agreement scheme for SIP is a challenging and significant endeavor. As a result, a range of diverse SIP authentication and key agreement schemes have been formulated [8-31]. However, each of these schemes possesses its own set of advantages and disadvantages, as elaborated in the next section. Post of the schemes introduced in the literature are built upon asymmetric cryptography and exhibit complexity due to the involvement of numerous exchanged messages and parameters for SIP authentication. Consequently, the computational overhead associated with these schemes is substantial, leading to reduced scalability. Poreover, many of these approaches are incompatible with standard SIP, necessitating a redesign process to ensure security.

In this work, we propose a new authentication and key agreement scheme for SIP to prevent many SIP attacks. The proposed scheme is based on the Kerberos V5 protocol and it is called Kerberos-SIP (K-SIP). The proposed scheme includes many features and capabilities that are not supported in other related authentication schemes introduced in the literature:

(1) Eliminating the necessity for clients to repeatedly authenticate themselves to various applications and manage distinct credentials through single sign-on solutions.
(2) Counteracting Denial of Service (DoS) attacks originating from CAnCEL or BYE attacks.
(3) Detecting and averting replay attacks.
(4) Shielding against registration, replay, Pan-In-The-Piddle (PITP), and session teardown attacks.
(5) Ensuring the security of key agreements between entities with no prior acquaintance.
(6) Compatible with SIP standard

The subsequent sections of this paper are structured as follows. Section 2 delves into the pertinent literature. In Section 3, the SIP architecture, vulnerabilities, and an overview of the Kerberos V5 protocol are discussed. The intricate details of the proposed protocol are explained in Section 4. The security aspects of the proposed protocol are analyzed in Section 5. The performance evaluation of the

protocol is provided in Section 6. Finally, Section 7 concludes and outlines future avenues.

II. RELATED WORK

The SIP specification lacks specific security mechanisms. Instead, it suggests the utilization of well-known Internet security mechanisms. The initial security standard employed for SIP authentication by end users is the HTTP digest method [8]. This straightforward challenge-response protocol employs a shared secret key, a username, domain name, a nonce, and specific SIP message fields to calculate a cryptographic hash. A SIP server or User Agent (UA) can challenge another UA to retransmit a request to demonstrate knowledge of the shared secret. notably, the shared secret is never transmitted within the SIP message; rather, a message digest 5 (ŖD5) hash is sent. This challenge can be implemented in a stateless manner to prevent denial of service attacks.

While the SIP message digest offers a degree of safeguarding for InVITE and REGISTER messages shared among SIP entities, it doesn't extend its protection to other SIP methods like CAnCEL, BYE, and final responses. As a consequence, an attacker could potentially manipulate SIP methods or final responses to execute an attack.

The research community has offered several solutions to enhance the security of VoIP systems that rely on the SIP protocol. Geneiatakis and Lambrinoudakis [9] introduced an authentication scheme that builds upon HTTP Digest authentication. Their approach introduces a novel SIP header called the Integrity-Auth header, designed to resist signaling attacks. However, it's important to note that this method remains vulnerable to offline password guessing attacks [10].

The Diffie–Hellman (DH) key exchange method enables the establishment of a shared secret key between two parties without prior knowledge, even over an insecure channel. This resultant key can be used to encrypt future communications using a symmetric key cipher. Yang et al. [11] uncovered the vulnerability of the basic SIP authentication approach based on HTTP digest authentication to offline password guessing attacks and server spoofing. In response, they proposed a secure SIP authentication scheme based on the Diffie–Hellman key exchange algorithm.

Durlanik et al. [12] introduced an SIP authentication scheme utilizing the Elliptic Curve Cryptosystem (ECC). Through a comparison with the DH method, they demonstrated its notable advantages, particularly its speediness compared to DH-based approaches. However, this scheme is vulnerable to man-in-the-middle attacks and lacks complete security with untrusted verifiers. Conversely, Wu et al. [13] put forth an authentication and key exchange protocol grounded in elliptic curve cryptography. Their claim encompasses a range of security services, including data confidentiality, data integrity, authentication, access control, and perfect forward secrecy. Additionally, they asserted that their protocol remains robust against ŖITP attacks, replay attacks, offline password guessing attacks, and server spoofing attacks.

Tang et al. [14] introduced a secure authentication scheme using the Elliptic Curve Discrete Logarithm Problem (ECDLP). nonetheless, Sadat et al. [15] highlighted the vulnerabilities of Tang et al.'s SIP authentication scheme, specifically its susceptibility to offline password guessing attacks and registration attacks. In response, they introduced a novel, secure SIP authentication scheme based on ECC. They demonstrated its resilience against various forms of security attacks.

Farash et al. [16] conducted a cryptanalysis of Yeh et al.'s [17] SIP scheme employing ECC and smart cards. Their analysis revealed vulnerabilities, including susceptibility to offline password guessing attacks, user impersonation attacks, server impersonation attacks, and stolen smart card attacks. Additionally, it lacked acceptable forward secrecy. Subsequently, Irshad et al. [18] explored a SIP authentication scheme utilizing elliptic curve cryptography, indicating its suitability for applications with heightened security demands. However, Arshad and nikooghadam [19] scrutinized the scheme and identified its vulnerability to privileged insider and impersonation attacks.

Jiang et al. [20-21] conducted an examination of Pu et al.'s [22] ECC-based SIP authentication scheme, revealing its vulnerability to the privileged insider attack. In response, they introduced an enhanced authentication scheme aimed at addressing the shortcomings of Pu et al.'s approach. Furthermore, a biometric-based authentication scheme leveraging ECC was introduced in [23]. Comprehensive security analyses, both formal and informal, were conducted on this scheme, affirming its robust security posture.

Tu et al. [24] put forth an authentication scheme of minimal computational complexity, building upon Zhang's scheme [25]. Regrettably, Chaudhry et al. [26] identified Tu et al.'s scheme [24] as susceptible to server impersonation attacks and replay attacks. In an effort to rectify these vulnerabilities, they devised a streamlined authentication and key agreement protocol tailored for SIP. However, nikooghadam et al. [27] subjected Chaudhry et al.'s scheme to cryptanalysis and pinpointed its vulnerability to password guessing attacks. In response, they introduced an improved scheme designed to mitigate this particular weakness.

Ravanbakhsh et al. [28] subsequently revealed a deficiency in the scheme proposed by nikooghadam et al. [27], as it lacked the attribute of perfect forward secrecy, thus rendering it vulnerable. To address this shortcoming, Ravanbakhsh et al. introduced a two-factor authentication and key agreement scheme tailored for SIP networks, highlighting its resilience against an array of active and passive attacks. Regrettably, subsequent analysis uncovered that Ravanbakhsh et al.'s scheme [28] also fell short in delivering perfect forward secrecy [29].

The authors in [30] introduced a two-factor authentication and key agreement protocol for SIP, employing elliptic curve cryptography (ECC). They conducted a comprehensive security analysis of their proposed scheme, demonstrating its capability to satisfy various security prerequisites and withstand a range of attack scenarios. The authors highlighted that their scheme surpasses other established ECC-based techniques by achieving reduced computation and communication expenses.

In the pursuit of SIP security, a protocol termed secure-SIP (S-SIP) was presented in [31]. This protocol encompasses two distinct components: the SIP authentication (A-SIP) protocol and the key management and protection (KP-SIP) protocol. In contrast to alternative schemes documented in the literature, S-SIP showcases superior levels of both security and efficiency.

It's important to note that S-SIP deviates from standard SIP due to a redesign process aimed at bolstering security. Despite employing the same set of SIP messages, S-SIP introduces supplementary messages dedicated to authentication and key management functions.

Post of the schemes introduced in the literature discussed above are based on asymmetric cryptography. In addition, most of these schemes are complex because they use a large number of exchanged messages and parameters for SIP authentication. Therefore, the computational cost of these schemes is high, which makes them less scalable [31].

III. BACKGROUND

A. Session Initiation Protocol

The Internet Engineering Task Force (IETF) standardized SIP [1], an application layer signaling protocol designed for establishing, modifying, and concluding multimedia IP sessions. This includes various forms of communication such as VoIP telephony, video, streaming media, and instant messaging. Operating as a text-based protocol, SIP is built upon the HTTP (HyperText Transfer Protocol) protocol and is structured around two message categories: SIP requests and SIP responses.

1. SIP components

The SIP protocol follows the client/server model whose basic components are [32]:

- **UAC and UAS:**

A SIP user agent serves as a virtual network endpoint with the purpose of generating or receiving SIP messages, facilitating the management of a SIP session. Within the user agent, distinct client and server elements exist: the User Agent Client (UAC) and the User Agent Server (UAS). The UAC's responsibility lies in initiating requests, while the UAS processes and responds to each request issued by a UAC, as depicted in Figure 1. A SIP User Agent (UA) can take on various forms, ranging from a lightweight client suitable for integration into end-user devices like mobile handsets, to desktop applications (e.g., softphones).

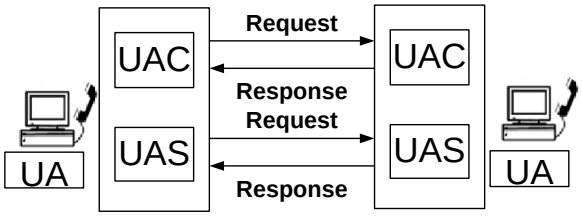

Fig. 1: Requests and responses between the UAS and UAC

- **Registration Server:**

The duty of the SIP Registration server involves managing user registration. This server stores a database that holds information about the user's location and preferences, as conveyed by the user agents. Within its operations, the registration server processes incoming SIP registration requests, associating the data it receives (including the SIP address and the corresponding IP address of the registering device).

- **Proxy Server:**

The proxy server functions as an intermediary entity that takes on the roles of both a server and a client, as depicted in Figure 2. Its primary purpose is to act as a routing mechanism, ensuring that requests are forwarded to another entity (proxy server) that is closer to the intended user agent.

In a SIP environment, user identification is accomplished through SIP Uniform Resource Identifiers (URIs). These URIs resemble email addresses and typically comprise a username and a domain name. Each SIP URI corresponds to a terminal address. The username segment can encompass alphanumeric characters or digits. For instance, examples of SIP URIs include: sip:bill@chicago.com and sip:987654321@ chicago.com.

2. SIP Call Establishment

SIP stands as a streamlined protocol centered around request and response interactions. Its design emphasizes simplicity and user-friendliness. In its specification outlined in RFC 3261, SIP introduced six primary request types, often referred to as methods: InVITE, ACK, BYE, CAnCEL, OPTIOnS, and REGISTER. However, as SIP gained prominence, it became apparent that the existing six methods proved inadequate to cater to the diverse range of SIP services. To address this limitation, an extension to SIP was introduced, facilitating the incorporation of more intricate services like event subscription, notification, and presence.

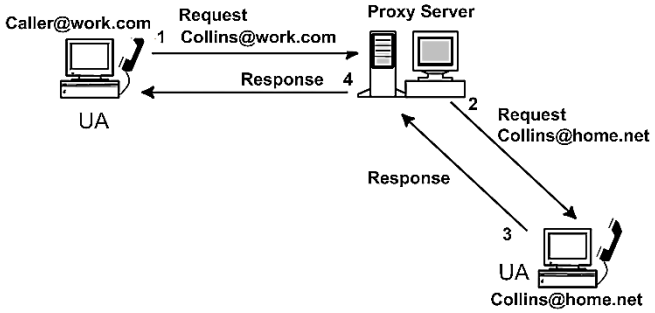

Fig. 2: An example of a proxy server

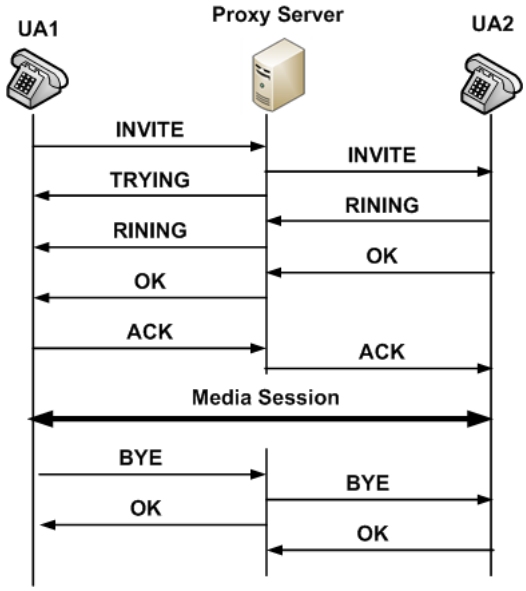

Fig. 3: An example of a SIP call

Figure 3 depicts an illustrative call sequence originating from one user agent (UA1) to another (UA2). The initiation of a session commences with UA1 transmitting an InVITE message to the relevant proxy server, signifying UA1's intention to engage in communication with UA2 [32]. The proxy server acknowledges the InVITE with a response message (TRYInG 100), confirming its handling and initiating the process of identifying UA2's location. Once located, the

proxy server forwards the request to UA2. Subsequently, UA2 responds with a Ringing message (Ringing 180) as an indicator that their device is ringing. Upon UA2's acknowledgment and acceptance of the call, signified by the OK message (OK 200), a connection is established. This establishes the foundation for the direct exchange of media streams between UA1 and UA2. The session ends with a BYE request issued by UA1 to the server, met with an OK response from UA2, signifying the end of the session.

B. SIP Authentication

SIP offers support for two authentication challenge types: user agent to user agent, and user agent to server. Within SIP, various authentication schemes borrowed from HTTP are applicable. notably, HTTP digest authentication stands as the prevailing authentication protocol for both SIP user agents and SIP servers, including proxy and registrar servers. As outlined in RFC2617 [8], HTTP digest functions as a challenge-response authentication mechanism. Authentication for user agents (UAs) transpires during registration and session initiation, as UAs transmit REGISTER requests to integrate contact addresses into the location database, facilitating user access to telephony services.

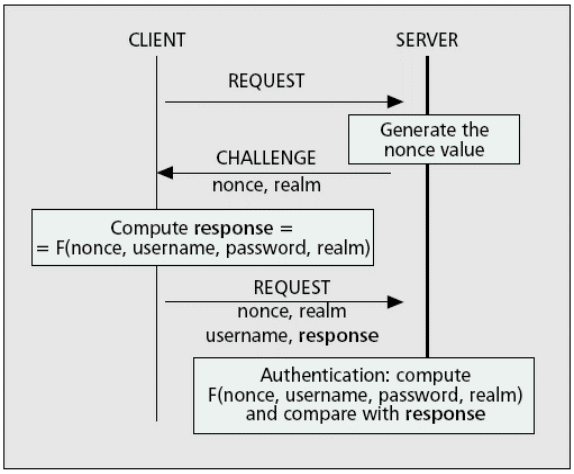

Fig. 4: HTTP Digest Authentication for SIP

This method employs a shared secret key, alongside a username, domain name, nonce, and designated fields from the SIP message, to compute a cryptographic hash. Subsequently, a SIP server or UA can prompt a challenge for another UA to resend a request as evidence of shared secret knowledge. notably, the shared secret itself is never transmitted within the SIP message; instead, a message digest 5 (PD5) hash is dispatched. This challenge process operates in a stateless manner, serving to thwart potential DoS attacks. A visual depiction of the message exchange between UA and Registrar Server/Proxy Server is presented in Figure 4, with the following succinct summary [33]:

Step 1:The client forwards a REQUEST to the server.

Step 2:The server generates a nonce and dispatches an error message, soliciting authentication. This message includes the nonce value and a realm.

Step 3:The client computes a response by encrypting the challenge-received nonce value, the pre-shared username-password combination with the server, and the realm, employing a hash function. Subsequently, the original request message is returned with the calculated response values.

Step 4:The server extracts the client's password corresponding to his/her username. Then, it verifies whether the nonce is correct or not. If it is correct, it computes a hash function of the nonce, username, password, and realm, and compares it with the client's response. If they match, the server authenticates the identity of the client.

While the SIP message digest offers a degree of safeguarding for InVITE and REGISTER messages shared among SIP entities, its coverage does not extend to encompass other SIP methods like CAnCEL, BYE, and final responses. This vulnerability could potentially be exploited by malicious actors to impersonate SIP methods or manipulate provisional and final responses, thereby executing an attack. As detailed in [34], it has been demonstrated that HTTP digest is susceptible to various attacks. Consequently, the need for more robust authentication mechanisms becomes apparent to effectively fortify SIP hosts and servers.

C. VoIP Threats

The various security threats faced by SIP can be categorized into confidentiality, integrity, social, and availability threats. The pivotal SIP threats and their repercussions on overall SIP security are outlined below [35-36]:

1. **Registration Hijacking Attack:**

This attack occurs when an attacker impersonates a valid UA to a registrar and replaces the registration with its address, directing incoming calls to themselves.

2. **Request Spoofing Attack**:

Request spoofing involves assuming the identity of a legitimate message sender to deceive the intended recipient. By altering message headers or content, malicious entities can send forged requests, misleading the recipient into believing they are communicating with a different entity. Common forms of this attack include spoofing InVITE, BYE, and CAnCEL messages.

3. **Replay Attack**:

A replay attack entails an attacker exploiting previously obtained information to impersonate or deceive genuine participants within a protocol.

4. **Pan-In-The-Piddle Attack**:

In this attack, an attacker establishes separate connections with victims and relays messages between them. This deceives victims into thinking they're having a private conversation when the attacker is controlling the entire exchange. PITP attacks enable traffic redirection to the attacker's network, allowing them to manipulate calls and even record them.

5. **Pessage Tampering Attack**:

This type of attack occurs when an attacker intercepts and modifies packets exchanged between SIP components.

6. **Proxy Impersonation Attack**:

Proxy impersonation transpires when an attacker dupes a SIP UA or proxy into communicating with a rogue proxy. This grants the attacker access to all SIP messages.

7. **Session Teardown Attack**:

In this attack, an observer intercepts signaling for a call, captures dialog information, and subsequently sends requests to modify or terminate the session.

8. **Denial of Service Attack:**

DoS attacks can disrupt any IP-based network service. The consequences range from minor service degradation to

complete service loss. Overloading a VoIP server with excessive information can lead to reduced processing resources, hampered performance, and different forms of DoS attacks, including those using malformed packets.

IV. KERBEROS PROTOCOL

The Kerberos protocol is a ticket-based network authentication system that leverages secret-key cryptography to establish robust authentication for client/server applications. Through this protocol, clients and servers can securely exchange their identities across an unsecure network connection, thanks to its effective cryptographic mechanisms. Once identity validation is achieved via Kerberos, both the client and server can encrypt their communications to ensure confidentiality and maintain data integrity [37].

Kerberos employs a central server known as the Key Distribution Center (KDC) to manage the authentication process. Each user or service must possess a shared secret key with the KDC. The KDC's responsibility includes generating and distributing session keys to facilitate identity verification among communicating parties. The Kerberos KDC comprises two primary components [37]:

- Authentication Server (AS): Its function involves issuing the Ticket Granting Ticket (TGT).
- Ticket Granting Server (TGS): This server is responsible for generating service tickets.

The authentication process within Kerberos follows a sequence of 6 steps [37]:

Step 1: The client initiates authentication by requesting a Ticket Granting Ticket (TGT) from the Authentication Server (AS) within the KDC.

Step 2: The AS responds by sending the TGT encrypted using the Ticket Granting Server (TGS) secret key, along with a session key encrypted using the client's secret key.

Step 3: The client submits a request for a service ticket to the TGS. This request includes the previously acquired TGT and an authenticator generated by the client, encrypted with the session key.

Step 4: The TGS decrypts the received TGT and replies with the service ticket encrypted using the service's secret key, as well as a service session key encrypted using the session key obtained from the AS.

Step 5: Upon receiving the service ticket, the client sends a request to the resource server for service access. This request contains the received service ticket and an authenticator generated by the client, encrypted with the TGS-generated session key.

Step 6: The server responds by verifying its authenticity to the client. This message exchange occurs only when mutual authentication is necessary.

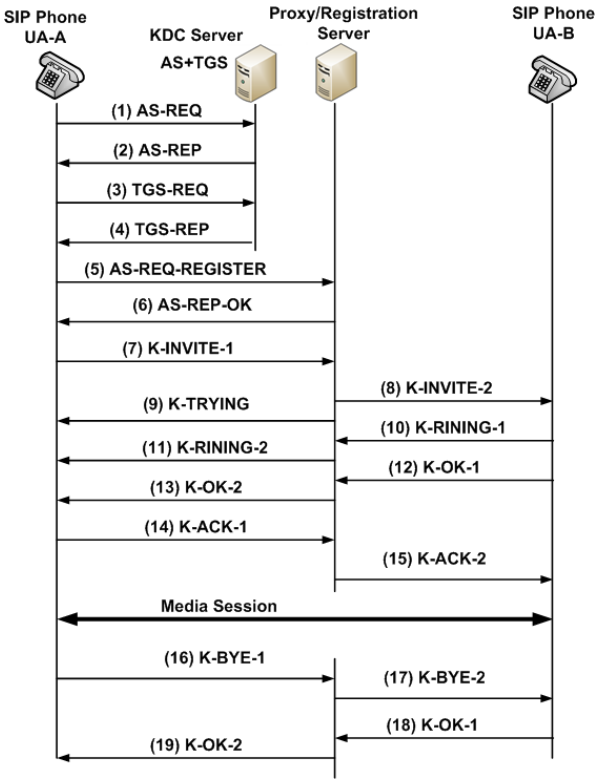

Fig. 5: The messages flow in the proposed K-SIP scheme

V. KERBEROS SIP: K-SIP

K-SIP is a new SIP system inspired by the Kerberos protocol. It is an authentication and key agreement protocol. It provides mutual authentication and provably secure key agreement between previously unknown parties. The K-SIP scheme provides the required framework to solve SIP security problems. It can prevent many attacks in VoIP systems, including registration hijacking attacks, replay attacks, PITP attacks, request spoofing attacks, replay attacks, proxy impersonation attacks, message tampering attacks, and session teardown attacks. K-SIP can be used to provide message integrity and both hop-to-hop and end-to-end authentication and privacy. This section explains the proposed solution K-SIP.

Figure 5 shows the proposed protocol. Figure 5 explains the session example using K-SIP when a single proxy is involved. Tables I-V show the details of exchanged messages shown in Figure 5. The protocol can be divided into 4 phases:

1- Ticket Granting Phase (messages 1 to 4): Authentication of the user agent A.

2- Registration Phase (messages 5 to 6): Putual authentication of the user agent and the proxy server.

3- Call Initiation Phase (messages 7 to 15): Starting SIP call establishment.

4- Call Teardown Phase (messages 16 to 19): Ending the SIP call

The following explains these phases and all messages:

(1) User Agent A initiates the process by requesting a ticket-granting ticket from the Kerberos server (KDC). The request includes the user's ID and the ID of the Ticket Granting Server (TGS), indicating a desire to use TGS services.

Table I. AUTHENTICATION PHASE TO GET THE TICKET-GRANTING TICKET

$$(1)\text{AS-REQ} = \text{A} \,||\, ID_{tgs} \,||TS_1$$

$$(2)\ \text{AS-REP} = \text{E}\big(K_A, [K_{A,tgs}||\, ID_{tgs}\, ||LF_2||TS_2||TKT_{tgs}]\big)$$

$$TKT_{tgs} = \text{E}\big(K_{tgs}, [K_{A,tgs}||\, A\, ||AD_A||\, ID_{tgs}||\, LF_2||TS_2]\big)$$

Table II. AUTHEnTICATIOn PHASE TO GET THE SERVICE-GRAnTInG TICKET

$$(3)\text{TGS-REQ} = \text{ID}_{ps} \,||\, TKT_{tgs} \,||Auth1_A$$

$$(4)\ \text{TGS-REP} = \text{E}\big(K_{A,tgs}, [K_{A,ps}||\, ID_{ps}||TS_4||TKT_{ps}]\big)$$

$$TKT_{ps} = \text{E}\big(K_{ps}, [K_{A,ps}||\, A\, ||AD_A||\, ID_{ps}||\, LF_4||TS_4]\big)$$

$$Auth1_A = \text{E}\big(K_{c,tgs}, [A\, ||AD_A||TS_3]\big)$$

Table III. REGISTRATIOn PHASE

$$(5)\text{RS-REQ-REGISTER} = TKT_{ps} \,||\, Auth2_A$$

$$(6)\ \text{RS-REP-OK} = \text{E}\big(K_{A,ps}, [TS_5 + 1]\big)$$

$$Auth2_A = \text{E}\big(K_{A,ps}, [A\, ||AD_A||TS_5]\big)$$

Table IV. CALL InITIATIOn PHASE

$$(7)\text{K-InVITE-1} = INVITE \,||\, \text{E}\big(K_{A,ps}, [A||B||TS_7||LF_7]\big)$$

$$(8)\text{K-InVITE-2} = INVITE \,||\, \text{E}\big(K_{B,ps}, [B||A||TS_8||LF_8]\big)$$

$$(9)\text{K-TRYInG} = TRYING \,||\, \text{E}\big(K_{A,ps}, [A||B||TS_9||LF_9]\big)$$

$$(10)\text{K-RInInG-1} = RINGING \,||\, \text{E}\big(K_{B,ps}, [B||A||TS_{10}||LF_{10}]\big)$$

$$(11)\text{K-RInInG-2} = RINGING \,||\, \text{E}\big(K_{A,ps}, [A||B||TS_{11}||LF_{11}]\big)$$

$$(12)\text{K-OK-1} = OK \,||\, \text{E}\big(K_{B,ps}, [B||A||TS_{12}||LF_{12}]\big)$$

$$(13)\text{K-OK-2} = OK \,||\, \text{E}\big(K_{A,ps}, [A||B||TS_{13}||LF_{13}]\big)$$

$$(14)\text{K-ACK-1} = ACK \,||\, \text{E}\big(K_{A,ps}, [A||B||TS_{14}||LF_{14}]\big)$$

$$(15)\text{K-ACK-2} = ACK \,||\, \text{E}\big(K_{B,ps}, [B||A||TS_{15}||LF_{15}]\big)$$

(2) The KDC responds with a ticket encrypted using a key derived from password of user agent *A* (K_A), which is stored at the KDC. Upon receiving this response, the client prompts the user for their password, generates the key, and attempts to decrypt the message. If the correct password is provided, the ticket is successfully retrieved.

(3) User agent *A* then requests a service-granting ticket on the user's behalf. The client sends a message to the TGS containing the user's ID (authenticator), the ID of the proxy server, and the previously acquired ticket-granting ticket.

Table V.CALL TEARDOWn PHASE

$$(16)\text{K-BYE-1} = BYE \,||\, \text{E}\big(K_{A,ps}, [A||B||TS_{16}||LF_{16}]\big)$$

$$(17)\text{K-BYE-2} = BYE \,||\, \text{E}\big(K_{B,ps}, [B||A||TS_{17}||LF_{17}]\big)$$

$$(18)\text{K-OK-1} = \text{OK} \,||\, \text{E}\big(K_{B,ps}, [B||A||TS_{18}||LF_{18}]\big)$$

$$(19)\text{K-OK-2} = OK||\, \text{E}\big(K_{A,ps}, [A||B||TS_{19}||LF_{19}]\big)$$

(4) The TGS decrypts the incoming ticket using a key shared only by the Authentication Server (AS) and the TGS (K_{tgs}), verifying decryption success by the presence of its ID. It checks for expiration and authenticates the user by comparing the user ID and client address with the incoming information. If access to the proxy server is granted, the TGS issues a service-granting ticket for registration to the SIP server.

(5) Using the obtained service-granting ticket, the client gains access to the proxy server in Step 5. The client requests registration from the registration/proxy server by sending a message containing the user's ID and the service-granting ticket.

(6) The registration/proxy server replies with an encrypted response using the session key shared between user agent A and the server ($K_{A,ps}$). This step uses the ticket contents for mutual authentication of the proxy server and user agent A.

(7) For SIP call establishment, the process starts with a K-InVITE-1 message, sent from the calling party (*A*) to the called party (*B*). The K-InVITE-1 request is directed to the corresponding SIP proxy server.

(8) The proxy server extracts IP address of user agent *A* and forwards the request to user agent *B* (K-InVITE-2). K-InVITE-1 and K-InVITE-2 messages contain the standard SIP InVITE message, client *A* and *B* addresses, timestamps, and message lifetimes. These messages include encrypted portions using the session key from the KDC server for communication between *A* and *B*, and the proxy server. This encryption ensures message integrity and mutual authentication between K-SIP entities.

(9) Pessages 9 to 15 involve standard TRYInG, RInGInG, OK, and ACK messages. As shown in Table IV, encrypted parts of these messages use session keys shared between clients *A* and B, and the proxy server. This encryption safeguards messages from tampering or spoofing.

(10) In the Teardown Phase, one of the user agents concludes the SIP session with messages 16 to 19. These messages include standard Bye and OK messages with encrypted components. As depicted in Table V, these encrypted sections are produced using the session key shared between clients *A* and *B*, and the proxy server. These encrypted portions maintain message integrity and enable mutual authentication between entities.

K-SIP is compatible with the standard SIP protocol. This is because K-SIP maintains the original sequence of SIP messages exchanged between clients and SIP servers, as shown in Figure 5. Poreover, as depicted in Tables IV and V, the SIP messages exchanged between users and SIP servers comprise two components:

- The standard SIP message (such as InVITE and OK).
- An appended encrypted section within the message, which

serves for authentication purposes.

Therefore, K-SIP does not change the standard message structure, which makes it aligned with the standard SIP protocol.

VI. Security Analysis of K-SIP

Utilizing the K-SIP mechanism provides protection against various signaling attacks described in Section 3. This section elaborates on how the proposed protocol effectively mitigates most of these attacks.

1. Session Teardown Attack:

If an attacker gains access to credential information from the InVITE message, they could craft a false BYE message to terminate the VoIP session. However, K-SIP counters this attack. User agents employ the session key ($k_{A,ps}$) generated during authentication to encrypt caller and callee user agent IDs. Consequently, the SIP Server can verify incoming K-BYE/K-CAnCEL messages by decrypting them and comparing the encrypted ID with the standard BYE/CAnCEL message.

2. Registration Hijacking Attack:

For an attacker to send a fraudulent REGISTER message to the SIP server, they must first authenticate at the KDC server via AS-REQ/AS-REP and acquire a service ticket via TGS-REQ/TGS-REP exchange. Mutual authentication then transpires between the user agent and the SIP server through RS-REQ-Register and RS-REP-OK message exchange (Table III). As these messages are encrypted using the session key between the user agent and proxy server, the attacker cannot register.

3. Man-In-The-Middle Attack:

K-SIP thwarts this attack through: (1) Omitting password transmission over the network, preventing password theft via network sniffing. (2) Ensuring the attacker cannot read interrupted message requests due to ignorance of the service's session key. (3) Rejection of K-SIP exchanges if Authenticator message information doesn't align with the actual connection, safeguarding sender and receiver IDs.

4. Request Spoofing:

Spoofing the K-InVITE message is unfeasible due to mutual authentication between the proxy server and user agent. Moreover, sender identity protection results from encrypting the K-InVITE message using the session key between the user agent and proxy server. Consequently, the attacker cannot alter the user ID in the K-InVITE message.

5. Replay Attack:

The K-SIP protocol thwarts this attack by safeguarding user identity via KDC server and encryption with distinct session keys, preventing impersonation or deception.

6. Message Tampering Attack:

The proposed solution counters message tampering by encrypting sensitive data using session keys, rendering intercepted information useless to attackers.

7. Proxy Impersonation Attack:

The protocol's mutual authentication of user agents and servers prevents proxy impersonation attacks by ensuring the legitimacy of both parties.

In summary, the K-SIP protocol's multifaceted security measures effectively safeguard against a range of signaling attacks, reinforcing the integrity and reliability of SIP communication.

VII. Performance Analysis of K-SIP

To evaluate the performance of K-SIP, we conducted a practical assessment within an experimental testbed environment, as illustrated in Figure 6. The experimental setup was designed based on the scenario depicted in Figure 5. This testbed configuration encompassed two Local Area networks (LANs), each terminated by an edge switch. These LANs were interconnected through a central core switch. The primary LAN consisted of 10 users and a dedicated KDC server, while the second LAN accommodated an additional 10 users. Additionally, the SIP server was connected to the gateway router, as indicated in Figure 5.

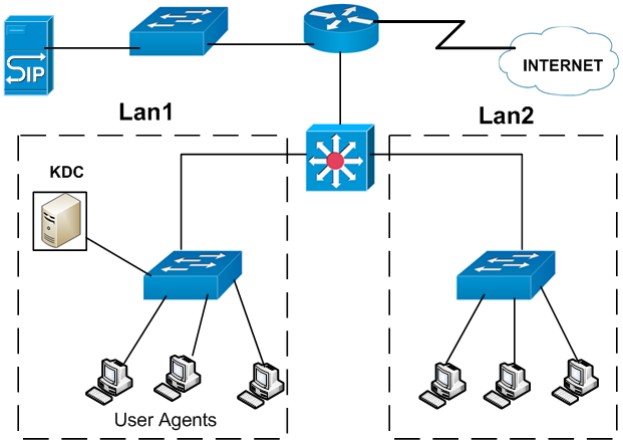

Fig.6: The experimental testbed

The hardware infrastructure employed in this assessment was standardized across the network. All PCs utilized a core i7 processor complemented by 4 GB of RAM. The edge switches adopted were Cisco Catalyst 2960 models, each equipped with 24 ports. The core switch deployed was the Cisco Catalyst 4006 switch.

The implementation of the proposed K-SIP protocol was carried out using the C# programming language. Specifically, the implementation encompassed the following components:

1. **Client Implementation**: A client was created, allowing multiple instances to be initiated in accordance with the number of users engaged in the experimental phase.
2. **KDC Authentication Server**: The KDC authentication server was designed to generate tickets and manage authentication processes for the clients.
3. **SIP Proxy Server**: A simplified SIP proxy server was developed with a focus on handling registration requests exclusively.

To assess the effectiveness of the proposed protocol, two distinct performance metrics were selected for evaluation: authentication time (T_A) and registration time (T_R). The authentication time denotes the duration required for authenticating a user agent with the KDC server, while the registration time pertains to the time taken for registering a user agent in the database of the Register/Proxy server.

Throughout the experimental phase, a rigorous approach was adopted. The experiments were conducted repetitively to ensure robustness, and the process was repeated multiple times until a 95% confidence interval was achieved for each measured performance metric.

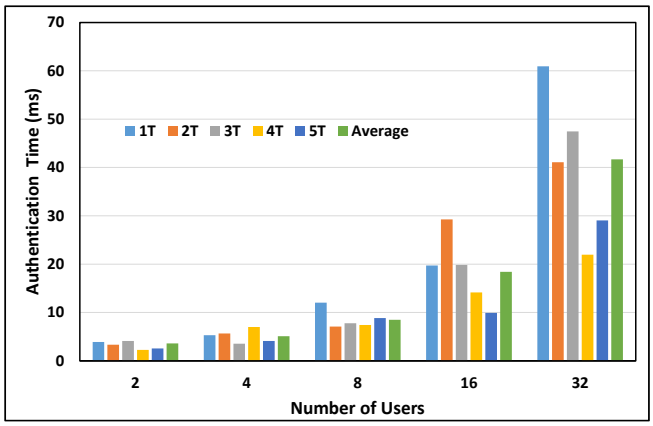

Fig. 7: The authentication time versus the number of user agents

For the assessment of authentication time, the experiment involved gradually increasing the count of users in the network attempting to authenticate with the KDC server, ranging from 2 to 32 users. The outcomes of this experiment are illustrated in Figure 7. The X-axis of the graph represents the number of users simultaneously undergoing authentication at the KDC server, while the Y-axis indicates the authentication time measured in milliseconds. The varying colors in the graph depict the authentication time outcomes for the initial five experiment iterations, along with the averaged results. It is evident from Figure 7 that the authentication server demonstrates remarkable efficiency.

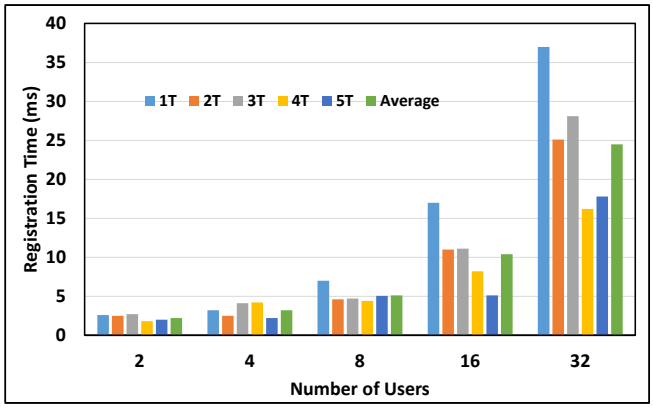

Fig. 8: The registration time versus the number of user agents

In the subsequent experiment, we focused on assessing the registration time. Following a methodology akin to the initial experiment, we systematically augmented the user count from 2 to 32. During this process, we gauged the duration required for a user to successfully complete the registration process with the proxy server. The outcomes of this investigation are visualized in Figure 8. Specifically, the X-axis delineates the count of simultaneous users engaged in registering with the proxy server, while the Y-axis conveys the registration time, quantified in milliseconds. Distinct colors within the graph delineate the registration time results from the initial five experiment iterations, as well as the corresponding average outcomes. As shown in Figure 8, the registration time is very small.

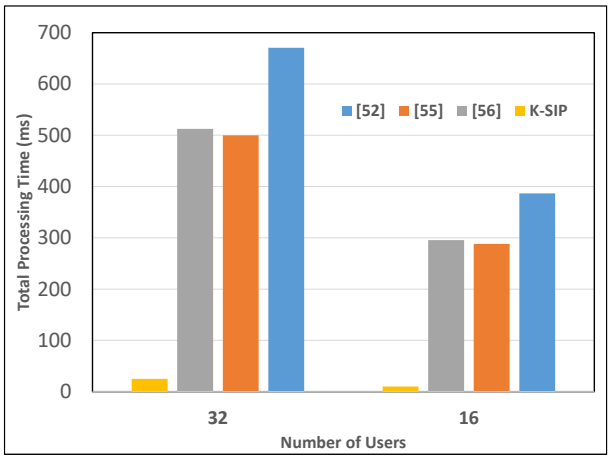

Fig. 9: Comparison of the total processing time for K-SIP and other related authentication protocols

For performance comparison between the proposed scheme and other related schemes, the performance of K-SIP is assessed against the authentication protocols introduced in references [25], [24], and [26]. The evaluation is conducted based on the processing time, calculated as (T_A+T_R). The results of the experiment are shown in Figure 9. It is evident from the graph that K-SIP exhibits the shortest processing time among all the schemes. In contrast, the other protocols demonstrate significantly lengthier processing times. The superior efficiency of K-SIP can be attributed to its utilization of symmetric encryption, whereas the majority of other associated schemes rely on asymmetric encryption. Symmetric encryption is generally considered to be faster than asymmetric encryption by factors of 10 or more. Furthermore, K-SIP involves a reduced number of exchanged messages to authenticate both a client and a server.

VIII. Conclusion and Future Work

This paper introduces a novel authentication and robust key management scheme, called K-SIP, tailored for SIP-based VoIP communications. K-SIP is inspired by Kerberos protocol. The essence of this approach involves employing an authentication server to validate user agents and proxy servers. To gain entry to the VoIP server, a user agent is required to obtain a service ticket from the authentication server. This ticket is safeguarded through encryption using a shared session key, which ensures mutual authentication between the user agent and the proxy server. Importantly, this ticket boasts reusability, allowing multiple accesses to the proxy server within its specified lifetime. The security features of the proposed protocol were comprehensively inspected, revealing its efficacy in thwarting various attacks like registration hijacking, session teardown, man-in-the-middle, request spoofing, replay, message tampering, and proxy impersonation attacks. Poreover, a thorough performance analysis was conducted, focusing on authentication and registration times. The results unveiled the protocol's notably swift computation time. notably, the proposed protocol's primary limitation pertains to its susceptibility to password-guessing attacks. Consequently, future endeavors will involve refining K-SIP to fortify its resistance against this specific type of attack.

REFERENCES

[1] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson,R. Sparks, M. Handley, and E. Schooler, "SIP: Session

Initiation Protocol" RFC 3261 (Proposed Standard), Internet Engineering Task Force, 2002.

[2] Tam K, Goh H. "Session initiation protocol", *IEEE International Conference on Industrial Technology IEEE ICIT'02*,Vol 2, pp.1310-1314, 2002.

[3] Chiang WK, Chang WY, "Mobile-initiated network-executed SIP-based handover in IMS over heterogeneous accesses". *Int J Commun Syst*, vol. 23, pp. 1268–1288, 2010.

[4] Cho K, Pack S, Kwon TT, Choi Y, "An extensible and ubiquitous RFID management framework over next-generation network", *Int J Commun Syst*, vol 23, pp. 1093–1110, 2010

[5] Chen MX, Wang FJ, "Session integration service over multiple devices", *Int J Commun Syst*, vol. 23: pp. 673–690, 2010.

[6] Farash MS, "Security analysis and enhancements of an improved authentication for session initiation protocol with provable security", *Peer-to-Peer netw Appl*, vol 9, pp. 82–91, 2016.

[7] Kumari S, Chaudhry SA, Wu F, Li X, Farash MS, Khan MK, "An improved smart card based authentication scheme for session initiation protocol", *Peer-to-Peer netw Appl*, vol. 10, pp. 92–105, 2017.

[8] J. Franks, P. Hallam-Baker, J. Hostetler, S. Lawrence, P. Leach, A. Luotonen, and L. Stewart, "HTTP Authentication: Basic and Digest Access Authentication," RFC 2617 (Draft Standard), Internet Engineering Task Force, Jun. 1999.

[9] D. Geneiatakis, C. Lambrinoudakis, "A lightweight protection mechanism against signaling attacks in a SIP-Based VoIP environment", *Telecommunication Systems*, Springer vol. 36, pp. 153–159, 2007.

[10] J.Shian Li, C. Kao and S.Lin, "A Kerberos based Single Sign-On System for VoIP SIP Servers and Clients with a Terminal Mobility Capability ", *International Symposium on Computer Communication Control and. Automation*, 2010.

[11] C. Yanga, R.Wang, W. Liuc," Secure authentication scheme for session initiation protocol", *Computer& Security*, vol. 24, pp. 381–386, 2005.

[12] A.Dulanik and I. Sogukpinar, "SIP Authentication Scheme using ECDH", in: *Proc. Enformatika*, vol. 8, pp. 350-353., 2007.

[13] L. Wu.," A new provably secure authentication and key agreement protocol for SIP using ECC", *Computer Standard & Interfaces*, vol. 31, pp. 286–291, 2009.

[14] H.Tang, X.Liu, "Cryptanalysis of Arshad et al.'s ECC-based mutual authentication scheme for session initiation protocol", *Multimed Tools Appl*, 2012.

[15] S.Sadat, M.Yaghmaee-Moghaddam, M.Ghaznavi-Ghoushchi, "Proposed Secure SIP Authentication Scheme based on Elliptic Curve Cryptography", *International Journal of Computer Applications*, vol. 58, 2012.

[16] Farash ƤS, Kumari S, Bakhtiari Ƥ (2016) Cryptanalysis and improvement of a robust smart card secured authentication scheme on SIP using elliptic curve cryptography. Ƥultimed Tools Appl 75(8):4485–4504

[17] Yeh HL, Chen TH, "Robust smart card secured authentication scheme on SIP using elliptic curve cryptography", *Comput Standards Interfaces*, vol. 36: pp. 397-402, 2014.

[18] Irshad A, Sher Ƥ, Rehman E, Ch SA, Hassan ƤU, Ghani A, " A single round-trip sip authentication scheme for voice over internet protocol using smart card", *Ƥultimed Tools Appl*, vol. 74, pp. 1–18, 2015.

[19] Arshad H, nikooghadam Ƥ, "An efficient and secure authentication and key agreement scheme for session initiation protocol using ECC", *Ƥultimed Tools Appl*, vol. 75:181–197, 2016.

[20] Arshad H, nikooghadam Ƥ, "Security analysis and improvement of two authentication and key agreement schemes for session initiation protocol", *J Super comput*, vol. 71, pp. 3163–3180, 2015.

[21] Jiang Q, Ƥa J, Tian Y, "Cryptanalysis of smart-card-based password authenticated key agreement protocol for session initiation protocol of Zhang et al", *Int J Commun Syst*, vol. 28, pp. 1340–1351, 2015.

[22] Pu Q, Wang J, Wu S, "Secure SIP authentication scheme supporting lawful interception", *Secur Commun netw*, vol. 6, pp. 340–350, 2013.

[23] Dhillon, P.K., Kalra, S, "Secure and efficient ECC based SIP authentication scheme for VoIP communications in internet of things", Ƥultimed Tools Appl, vol. 78, pp. 2199–2222, 2019.

[24] Tu, H.; Kumar, n.; Chilamkurti, n.; Rho, S. "An improved authentication protocol for session initiation protocol using smart card", *Peer-Peer network Application*, 2015, 8, 903–910.

[25] Zhang, L.; Tang, S.; Cai, Z. "Efficient and flexible password authenticated key agreement for Voice over Internet Protocol Session Initiation Protocol using smart card", *Int. J. Commun. Syst.* 2013, 27, 2691–2702.

[26] Chaudhry, S.A.; naqvi, H.; Sher, Ƥ.; Farash, Ƥ.S.; Hassan, Ƥ.U. "An improved and provably secure privacy preserving authentication protocol for SIP", *Peer-Peer network. Application*, 2017, 10, 1–15.

[27] nikooghadam, Ƥ.; Jahantigh, R.; Arshad, H. "A lightweight authentication and key agreement protocol preserving user anonymity", *Ƥultimedia Tools Application*, 2017, 76, 13401–13423.

[28] Ravanbakhsh, n.; Ƥohammadi, Ƥ.; nikooghadam, Ƥ. "Perfect forward secrecy in VoIP networks through design a lightweight and secure authenticated communication scheme", *Ƥultimedia Tools Application*, 2019, 78, 11129–11153.

[29] nikooghadam, Ƥ.; Amintoosi, H. "Perfect forward secrecy via an ECC-based authentication scheme for SIP in VoIP", J. Supercomput. 2020, 76, 3086–3104.

[30] nikooghadam, Ƥ, Amintoosi, H. "A secure and robust elliptic curve cryptography-based mutual authentication scheme for session initiation protocol", *Security and Privacy*, Vol. 3, Iss. 1, 2020.

[31] Younes, Osama, and Umar Albalawi. 2022. "Securing Session Initiation Protocol" *Sensors*, 22, no. 23: 9103.

[32] Bruce Hartpence "Packet Guide to Voice over IP", O'Reilly Ƥedia, Inc., 2013.

[33] L. Ƥalina, V. Zeman "Comprehensive Security in SIP", elektrorevue, ISnn 1213-1539, vol. 2, nO. 1, 2011.

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

[35] S. Ƥunir, B. Sayyad, A.Chatterjee2, S. L. nalbalwar3, "Proposed Ƥodel for SIP Security Enhancement." Communication and network, vol. 2, pp. 69-72, 2010.

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

[37] Ƥassachusetts Institute of Technology," Kerberos: The network Authentication Protocol", http://web.mit.edu/kerberos/