Paper: ASAT-15-109-CM

15th International Conference on
AEROSPACE SCIENCES & AVIATION TECHNOLOGY,
ASAT - 15 – May 28 - 30, 2013, Email: asat@mtc.edu.eg ,
Military Technical College, Kobry Elkobbah, Cairo, Egypt,
Tel: +(202) 24025292 –24036138, Fax: +(202) 22621908

SIP Security: Main Vulnerabilities, Denial of Service (DoS) Attacks and Intrusion Detection Techniques

D. Allawi*, A. E. Rohiem†, A. El-moghazy‡, and A. Ghalwash§

Abstract: Session Initiation Protocol (SIP) is application layer signaling text-based protocol used for creating, modifying, and terminating multimedia communications sessions (Internet telephone calls, instant messaging, and multimedia conferences) among Internet endpoints. SIP is defined by the Internet Engineering Task Force (IETF) and documented in RFC 3261. Unfortunately, SIP-based application services using IP network are not only exposed to the security vulnerabilities inherited from IP but also exposed to new security vulnerabilities inherited from SIP.
In this paper we present the most important security vulnerabilities, threats, and attacks against SIP- multimedia communications systems. Our goal is to provide roadmap to the interested persons for understanding existing capabilities, and identifying the gaps and vulnerabilities in SIP, We illustrate how these vulnerabilities can be exploited to compromise the security of SIP-based systems. Then we focus on Denial of Service (DoS) attacks that impact service availability along with the main detection techniques for these attacks.

Keywords: Session initiation protocol, SIP security, denial of service attacks, intrusion detection systems.

1. Introduction

To understand different security issues, we present the proper definitions in the context SIP-based systems security:
Vulnerability: a flaw or weakness in a system design, implementation, or management that could be exploited to violate the system security policy.
Intrusion: any attempt to compromise the integrity, confidentiality and availability of a resource can be categorized as Intrusion.
Denials of service (DoS) attacks: are attacks that deny the use of resources to legitimate users of the system information or capabilities.
Internet is susceptible to a plethora of attacks and undoubtedly it must be considered as a hostile environment by every critical real-time application such as SIP-based systems. Thus, the deployment of various SIP-based systems services raises much security challenges. On the contrary, the open architecture of SIP-based systems makes these services vulnerable not only to well known Internet attacks but also to more sophisticated attacks aiming to exploit vulnerabilities that may exist in the signaling or the media transport of SIP-based systems infrastructures.

* dahham78@hotmail.com
† Egyptian Armed Forces, Egypt; alaa_rohiem@yahoo.co.uk
‡ moghazymtc@yahoo.com
§ Helwan University, Cairo, Egypt, atef_ghalwash@yahoo.com

SIP is used for many session-oriented applications, such as calls, multimedia distributions, video conferencing, presence service and instant messaging. Major standards bodies including 3GPP, and ITU-I have all adopted SIP as the core signaling protocol for Next Generation Networks predominately based on the Internet Multimedia Subsystem (IMS) architecture [1].
SIP-multimedia communications systems have become widely deployed and developed rapidly. This rapid development in these systems brings with it new and much security threats. SIP is example of the open interfaces that can be used to attack systems. This paper introduces SIP security problems, focusing on SIP security. Section 2 presents overview of the SIP architecture. Section 3 describes major vulnerabilities in SIP. Section 4 addresses the possible threats and attacks against SIP-based systems. Section 5 focuses on DoS of SIP and its detection systems. While section 6 concludes the paper providing some pointers to future work.

2. SIP Overview

SIP is an application-layer protocol standardized by IETF, and is designed to support the setup of bidirectional communication sessions. It is somewhat similar to HTTP, in that it is text-based, has a request-response structure, and uses a user authentication mechanism based on the HTTP Digest Authentication. However, it is an inherently stateful protocol that supports interaction with multiple network components (e.g., PSTN bridges), and can operate over UDP, TCP, and SCTP [1] although it is more commonly operating over UDP.
All SIP messages are either requests from a client or responses to the request from the server [1]. SIP requests are also called methods; Table 1 shows the basic ones. Other methods such as Refer and Notify are proposed as extensions for the original methods.

Table 1. Basic SIP methods

Seq.	Method	Description
1	INVITE	Let invite a user or a service to a new session or to modify parameters of an established session.
2	ACK	Confirm the session establishment.
3	OPTION	Request information about the capabilities of a server.
4	BYE	End of a session.
5	CANCEL	Cancel a pending request.
6	REGISTER	Register the user agent.

For each request SIP server generates SIP response to indicate the status of the request. Each response message is identified by a numeric status code, Table 2 summarize these responses.

SIP is a client-server protocol, the main SIP entities are endpoints (soft phones or physical devices), a proxy server, a registrar, a redirect server, and a location server. Endpoints communicate with a registrar to indicate their presence. This information is stored in the location server. During call setup, the endpoint communicates with the proxy, which uses the location server to determine where the call should be routed to. This may be another endpoint in the same network or another proxy server in another network. Alternatively, endpoints may use a redirect server to directly determine where a call should be directed to, since redirect servers consult the location server in the same way that proxy servers operate during call setup. Once an end-to-end channel has been established between the two endpoints, SIP negotiates the session parameters (codecs, Real time Transmission Protocol (RTP) ports, etc.) using the Session Description Protocol (SDP). In a two-party call setup between Alice and Bob, Alice sends an INVITE message to her proxy server, optionally containing session

Table 2. SIP responses

Seq.	Response	Description
1	1xx Informational (provisional)	Request received, continuing to process the request.
2	2xx Success (final)	The action was successfully received, understood, and accepted.
3	3xx Redirection (final)	Further action needs to be taken in order to complete the request.
4	4xx Client Error (final)	The request contains bad syntax or cannot be fulfilled at this server.
5	5xx Server Error (final)	The server failed to fulfill an apparently valid request.
6	6xx Global Failure (final)	The request cannot be fulfilled at any server.

parameter information encoded within SDP. The proxy forwards this message directly to Bob, if Alice and Bob are users of the same domain. If Bob is registered in a different domain, the message will be relayed to Bob's proxy, and thence to Bob. While the call is being set up, Bob is sending RINGING messages. Once the call has been accepted, an OK message is sent to Alice, containing Bob's preferred parameters encoded within SDP. Alice responds with an ACK message. Following this exchange, the two endpoints can begin transmitting voice, video or other content using the agreed-upon media transport protocol, typically RTP. While the signaling traffic may be relayed through a number of SIP proxies, the media traffic is exchanged directly between the two endpoints. Figure 1 shows SIP multimedia connection establishment. When bridging different networks, e.g., PSTN and SIP, media gateways may disrupt the end-to-end nature of the media transfer to translate content between the formats supported by these networks.

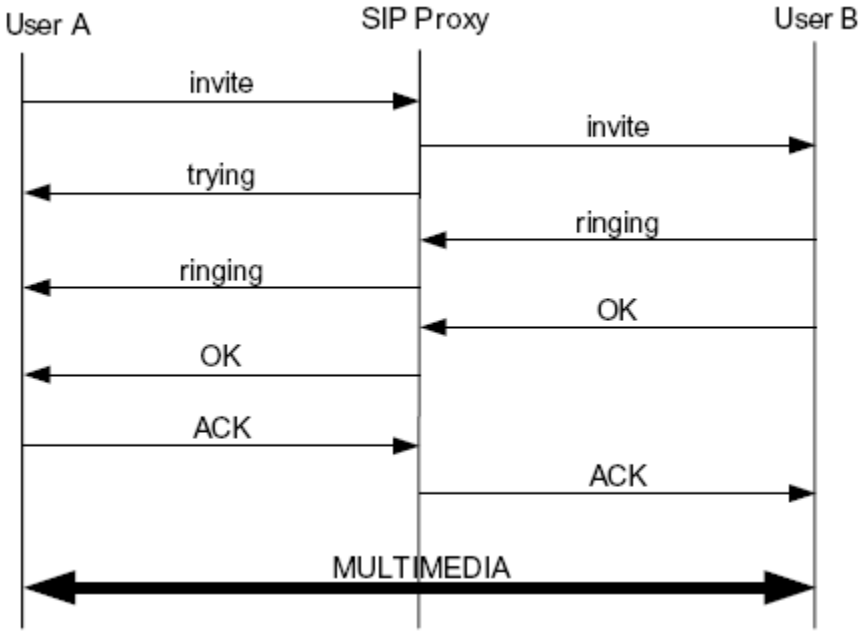

Fig. 1 SIP multimedia connection estableshment

3. SIP Vulnerability

This section focuses on the actual vulnerabilities in SIP, meaning the flaws that allow a threat agent to take advantage. Since the best way to eliminate security threats is to find and fix the vulnerabilities.

Vulnerabilities can be categorized based on different criteria. Therefore we will propose new criterion in categorization, this criterion depends on source of error, where we find the following types of vulnerabilities as shown in figure 2:

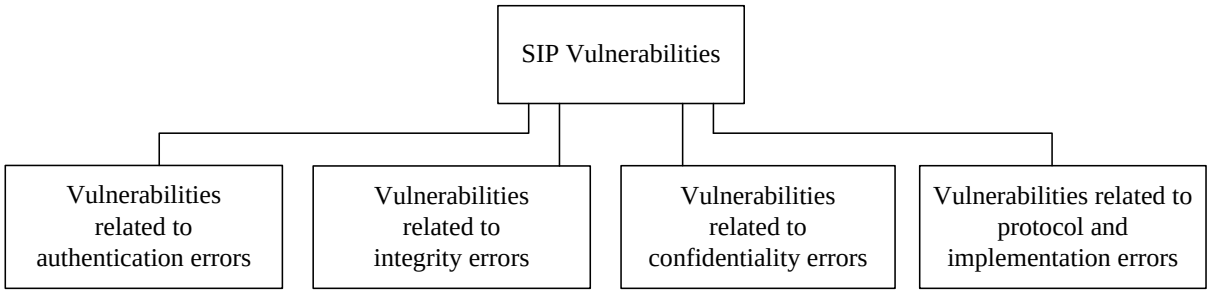

Fig. 2 Types of SIP vulnerabilities

A- Vulnerabilities related to authentication errors

Authentication is particularly difficult to achieve in SIP, since there are a number of intermediate elements such as proxies which possibly modify the contents of a message before it reaches the desired destination. All such intermediate elements must be trusted.

SIP Digest features several major weaknesses which can be easily exploited, vulnerabilities found in SIP based authentication are mentioned hereafter [2]:

- SIP authentication is applied to a few SIP messages (e.g., INVITE, REGISTER), and it leaves other important SIP messages (e.g., TRYING, RINGING, BYE, 200 OK, and ACK) unprotected. Where SIP servers and clients will process a BYE request without asking any authentication, BYE request is implicitly authenticated if it is received from the same network element (on the same path) as a previous INVITE. A third party attacker can thus observe the parameters of an eavesdropped INVITE message, and then insert a BYE request into the session. Once the BYE request is received by the target, the session would be torn down permanently. Similar attacks can be launched on RE-INVITE messages used to change session parameters.
- Deregistration is carried out by same authentication in the registration transaction, a wide variety of denial of service attacks also becomes possible if registration requests are not properly authenticated by registrars. If a malicious user is able to de-register some or all other users in the network and register his own device on their behalf, he can easily deny access to any of those users or services. Attackers can also try to deplete storage resources of the registrar by creating a huge number of bindings.
- SIP authentication protects a few SIP fields (e.g., *URL*, *username*, *realm*), and it leaves other important SIP fields (e.g., *SDP*, *From*, and *To*) in unprotected format.
- SIP authentication applied only to SIP messages from the client to the servers, and it leaves all the SIP messages from the SIP servers to client unprotected [3].
- SIP registration does not require the *From* field of a message to be the same as the *To* header field of the request, allowing third parties to change address-of-record bindings on behalf of another user. If the attacker can successfully impersonate a party authorized to change contacts on behalf of a user, he can arbitrarily modify the address-of-record bindings for the associated *To* address. Since SIP authentication relies implicitly on the authenticity of the server and intermediate proxies, the attacker who is able to successfully impersonate a server or a proxy can do arbitrary damage including denying service to the client or launching a

(distributed) denial of service attack. This requires the existence of some methodology for the client to authenticate the server or the proxy. Unfortunately, no such mechanism is specified in the SIP RFC.
- Due to the transactional model in SIP the request methods CANCEL and ACK are weakly authenticated. This is more or less impossible, since these methods operate in hop-by-hop mode and thus may be generated by any instance (server) in the signaling chain. It is improbable that every server has a security association with other instances, making authentication of these requests is effectively impossible. Also, the sequence numbers of these two request methods must be the same as the one of the requests to which they relate and thus cannot be challenged (leading to incrementing the number and thus not matching with the original message). This lack of authentication of CANCEL and ACK enables attackers to carry out injection attacks. An attacker can fake a CANCEL request resulting in a denial of session establishment. He can create a malicious ACK message (credentials in an ACK message are identical with those of the previous request) [4].
- Once a session has been established by initial messaging, subsequent requests (RE-INVITE) can be sent to modify the state of the session by same initial authentication. This procedure creates a new flaw to such requests to be forged by attackers.
- Most implementations accept the same credentials within a period of time, the attacker could replay messages (replay attack). In this context an attacker can register as a legitimate user. He is also able to redirect conversations to his device.
- Most of the devices do not check the source of the message. Attackers can infiltrate messages to manipulate or disturb SIP services. Also, flooding with connection requests to SIP clients (DoS attack) is likely. Established connections can be terminated and even directed to unauthorized instances.
- SIP includes an authentication mechanism based on the HTTP Digest mechanism. This authentication mechanism uses a challenge/response model. When the server receives the client response, it checks this response by repeating the MD5 calculation by using the stored value for password of user. If the calculated response is matched to the submitted response by the client, the request can be processed. The calculating the response is computationally costive task for the server, it has to look the user name, extracts the password from a database, combines this password with the original challenge and other information and then it calculates an MD5 checksum. An attacker can exploit this to run authentication flooding attack.

B- Vulnerabilities related to integrity errors
Two types are recognized hereafter:
- One of the weaknesses in SIP is the limited message integrity (the header is not included in the integrity calculation) [4]. An attacker can easily change the message or can be a MITM (man in the middle attack) sniffing valid credentials, change them and send it to a server.
- Generally, SIP parsers are being developed to receive and process well-formed messages, i.e. SIP messages conforming to the RFCs 3261 syntax [1]. However, an attacker, or even a poorly-implemented SIP client, is quite possible to generate and transmit various types of distorted messages [5], resulting to one of the undesired situations (denial of service, unstable operation, or unauthorized access).

C- Vulnerabilities related to confidentiality errors
The text-based nature of SIP messages gives more opportunities for attacks like spoofing, hijacking and message tampering in SIP applications, similarly to HTTP messages. The attacker can forge packets that manipulate device and call states. For example, such forged packets can prematurely terminate calls, redirect calls, or facilitate toll fraud.

D- Vulnerabilities related to protocol and implementation errors
These are listed hereafter:
- SIP has the same IP and application-level vulnerabilities [6]. It is well known that IP, which is used to transport SIP messages, is vulnerable to attacks like spoofing, session hijacking.
- Many SIP implementations still use the Universal Datagram Protocol (UDP) for transporting SIP messages. UDP is a connection-less, unreliable form of packet transfer. UDP does not use re-transmissions or sequence numbers, so it is easier for an attacker to spoof UDP packets. In contrast, the Transmission Control Protocol (TCP) is a connection oriented, guaranteed delivery transport. TCP is more secure than UDP, because it involves a negotiated setup and tear down, sequence numbers, and retransmissions for lost packets [6].
- The SIP-based application server employs SIP for signaling and the SIP protocol specification describes methods to end or terminate session, cancel an invitation, redirect a call and update session parameters. But SIP specification does not include any specific security mechanisms. It is very likely that attacker will try to exploit any security vulnerability in the SIP methods and cause DoS to the provided service.
- SIP protocol according to RFC 3261 utilizes transport protocols such as TCP, and UDP. As a result SIP inherits the vulnerabilities of these protocols. For instance, considering that the TCP is vulnerable to attacks like SYN flood or TCP session hijacking, it is highly likely that SIP will be also vulnerable to similar attacks.
- Another potential source of SIP security problems is that of SIP-based application bugs. Implementation flaws of SIP systems create opportunities for DoS attacks. A large number of systems are found to be vulnerable to malformed SIP messages [7].
- There is lack of expertise and security standards. Users might inadvertently expose the system.
- Until now SIP-based systems (for example VoIP) has been developed and deployed focusing on functionality with less thought for security [8]. Where there is not strong authentication in VoIP [9].

4. SIP Threats and Attacks

SIP-based systems suffer from all known attacks associated with any Internet application, as well as some of attacks specific to it. SIP-based network security attacks contain five main attacks [10, 11], as shown in figure 3:

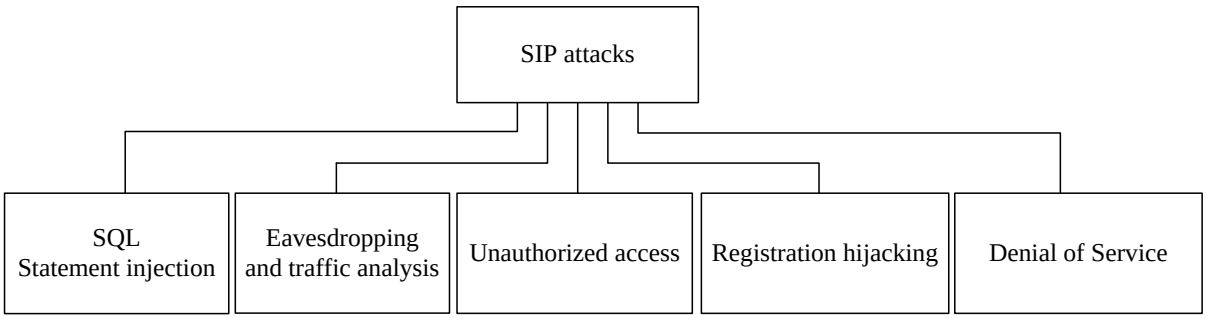

Fig. 3 Main SIP attacks

A- SQL statement injection
The text-based nature of SIP messages provides opportunity for message tampering attacks in SIP applications. SQL injection is kind of message tampering attack which already exploited successfully on Internet environment. The concept of SQL injection seems to be quite simple and can be launched in any application that creates and executes SQL statements. This attack

is not only targeting in data modification, but also in the downfall of database services to cause a DoS. SQL injection in SIP can be triggered every time to SIP network entity (e.g. SIP UA, SIP Proxy) is asking for authentication. So, in case a SIP network element requests authentication, the User Agent (UA) on behalf of the authorized user computers the appropriated credentials based on the HTTP Digest mechanism. The result of this computation (credentials) is included in the message's *authorization* header. Then the message is forwarded to the proxy server, which has to authenticate the received message. Thus, it recalculates user's credentials using the user's password stored in the subscriber table. To accomplish this task, it generates an SQL statement of the following syntax: " *SELECT* password *FROM* subscriber *WHERE* username='user A' *AND* realm = '192.168.1.13' ". In case a malicious user tries to launch an attack in the SIP architecture, exploiting SQL injection, he spoofs the SIP message and inserts the malicious SQL code in its *authorization* header. The code can be embodied in the username or in realm fields in the *authorization* header. As soon as the proxy receives a SIP message with an infected *authorization* header, it will generate and execute the dangerous SQL statement which may delete or modify data in the database [12].

B- Eavesdropping and traffic analysis

With SIP-multimedia connections, opportunities for eavesdroppers increase dramatically because of the large number of nodes in the path between two conservation entities. If the attacker compromises any of these nodes, he can access the IP packets flowing through that node. There are many free network analyzers and packet capture tools that can convert SIP-multimedia connection traffic to wave files [13]. These tools allow the attackers to save the conversation into the files and play them back on a computer [14].

C- Unauthorized access attack

Unauthorized access means that the attacker can access resources on a network that he does not have the authority [15]. Unauthorized access occurs when there are vulnerabilities in implementation issues [16]. The clear-text protocol exposes everything to anyone who can sniff the network traffic. The attacker might sniff the SIP traffic in local network to steal sensitive information. The use of malicious SIP messages by attacker is also a possibility and can cause unauthorized access or DoS. We will explain impersonation server as example on unauthorized access, as follows:

The destined domain by a request is generally specified in the Request-URI. UAs commonly contact a server in this domain directly in order to deliver a request. However, there is always a possibility that an attacker could impersonate the remote server, and that the UA's request could be intercepted by some other party. For example, consider a case in which a redirect server at one domain, one.com, impersonates a redirect server at another domain, two.com. A user agent sends a request to two.com, but the redirect server at one.com answers with a forged response that has appropriate SIP header fields for a response from two.com. The forged contact addresses in the redirection response could direct the originating UA to inappropriate or insecure resources, or simply prevent requests for two.com from succeeding.

Proxy impersonation occurs when an attacker tricks one of your SIP UAs or proxies into communicating with a rogue proxy. If an attacker successfully impersonates a proxy, he has access to all SIP messages and is in complete control of the call [6].

D- Registration hijacking [1]

The SIP registration mechanism allows a user agent to identify itself to a registrar as a device at which a user is located. A registrar assesses the identity asserted in the *From* header field of a REGISTER message to determine whether this request can modify the contact addresses associated with the address-of-record in the *To* header field. The *From* header field of a SIP

request, however, can be modified arbitrarily by the owner of a UA, and this opens the door to malicious registrations. An attacker that successfully impersonates a party authorized to change contacts associated with an address-of- record could, for example, de-register all existing contacts for a URI and then register their own device as the appropriate contact address, thereby directing all requests for the affected user to the attacker's device.
Registration hijacking occurs when an attacker impersonates a valid UA to a registrar and replaces the legitimate registration with its own address. This attack causes all incoming calls to be sent to the UA registered by the attacker [6].

E- Tearing Down Sessions [1]
Consider a case in which a third party attacker captures some initial messages in a session shared by two parties in order to learn the parameters of the session (*To tag*, *From tag*, and so forth) and then inserts a BYE request into the session. The attacker could opt to forge the request such that it seemed to come from either participant. Once the BYE is received by its target, the session will be torn down prematurely.
Similar mid-session threats include the transmission of forged RE-INVITEs that alter the session (possibly to reduce session security, modify media session, redirecting media to broadcast addresses can cause a DoS attack) [6].
Session tear down occurs when an attacker observes the signaling for a call, and then sends spoofed BYE messages to the participating UAs. Most SIP UAs do not require strong authentication, which allows an attacker to send a properly crafted BYE messages to the two UAs, tearing down the call [6].

F- Denial of Service (DoS) attacks
Denial of Service (DoS) attack is a serious threat for the Internet. DoS attacks can consume memory, CPU, and network resources and damage or shut down the operation of the resource under attack (victim). The aim of a DoS attack is to steal network resources, or to degrade the service perceived by users. Where this attack focuses on rendering a network of service unavailable.
The different types of denial of service attacks include: denial of access to information, denial of access to applications, denial of access to systems, and denial of access to communications. DoS is an issue for any IP network-based service, including electronic commerce, email, Domain Name Service (DNS), and SIP-multimedia connections (for example VoIP).
Because SIP-multimedia connections are other services on the IP network, it is just as susceptible to DoS as other IP network services. Plus, because it is a real-time service, it is even more susceptible to DoS attacks that impact delivery of audio and video. SIP creates a number of potential opportunities for DoS attacks since SIP entities open themselves to the public Internet in order to receive requests from worldwide IP hosts. DoS can take various forms, but generally involves an attack that prevents users from effectively using the targeted service. In next section we will explain DoS attack in more detail.

5. SIP Denial of Service and Intrusion Detection Systems (IDS)

5.1. SIP Denial of Service (SIP DoS)

SIP-multimedia connection is more widely deployed and as enterprises start to interconnect their internal networks via untrusted networks. For this reason, DoS consider is one of SIP-multimedia connection's most challenging threats. It is an issue now, and will become a more significant issue going forward. In this section we describe most significant denial of service attacks and its appropriate detection algorithms.

SIP DoS attack mechanisms differ according to attack type, some attacks exploit vulnerabilities in SIP protocol implementation, another utilize drawbacks exist in RFC protocol specification, where the others are resources consuming such as network bandwidth or agent processing capability [17]. We will divide SIP DoS attacks into four categories: Spoofed message attacks, flooding message attacks, malformed message attacks, and distributed DoS (DDoS), as shown in figure 4:

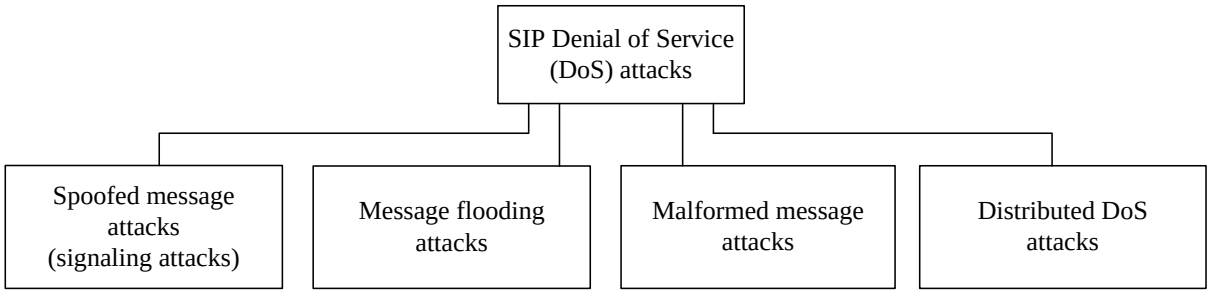

Fig. 4 Categories of SIP denial of service (DoS) attacks

5.1.1. Spoofed messages attacks (signaling attacks)

During call establishment, SIP agents exchange series of message, an attacker can impersonate himself as legal SIP client to modify, deny, or hijack SIP-multimedia calls. We will illustrate six important attacks in this category, as shown in figure (5):

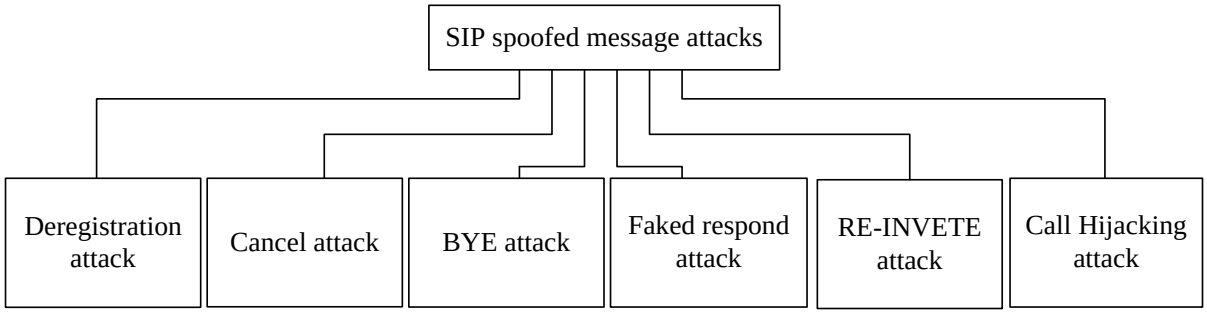

Fig. 5 Important SIP spoofed message attacks

A- Deregistration attack

A SIP REGISTER process informs the application server that a device is available to place and receive calls [1]. The register message includes both IP address and contact information of the user, along with the *expire* field, which gives the date and time after which the message content expires. The deregistration process accomplished by the same way, but the *expire* field is set to zero. In the De Registration Attack, the attacker sniff the network traffic, seek for registration message, and when found constructs spoofed message identical to the captured one except the *expire* field is set to zero, then direct it to the server. As a result the server removes the victim's record, and the victim has no indication that he isn't registered at the server [18].

B- Cancel attack

The CANCEL request is used to cancel a previous request sent by a client which server does not give final response yet. It asks the server to cease processing the request and to generate an error response to that request. Attacker listens on the network traffic for new calls and then terminates each call with a Cancel request. Attacker can authenticate himself using the same

authentication challenges in the original message request. A successful attack of this kind would quickly close down the call [19, 20].

C- BYE attack
SIP-multimedia calls are terminated by one of the call participants sending a SIP BYE request. Many SIP-multimedia application servers and clients process a BYE request without requiring authentication. An attacker easily constructs a BYE request and sends it to the server, which will then terminate the call [19].

D- Faked respond attack
SIP authentication applied is only to SIP messages from the client to the servers, and it leaves all the SIP messages from the SIP servers to client unprotected [3]. Attacker can easily exploit this vulnerability to sending a faked response to client, deny him from completing his call, or redirect the call to another callee. One example for this kind of attacks is USER BUSY attack. USER BUSY, is one possible response to the INVITE request. Attacker can send faked USER BUSY packets to prevent calls directed to specific callee.

E- Call Hijacking attack
Call hijacking attack refers to a situation where one of the intended end points of the conversation is exchanged with the attacker [19]. Once a call is hijacked, it is simple to forward it to the original callee, thus realizing a man in the middle (MITM) attack. Some of SIP methods can be used for call hijacking attack, as follows:
Using REFER method: A SIP REFER request is sent to a phone directs that phone to place a call to a supplied number or SIP URI [21]. A REFER request can be spoofed easily by attacker as any SIP request. A REFER can even be sent to a phone that does not have an active call [22].
Using REGISTER method: This method relies on two successive processes, deregistration and registration. Attacker sniffs the target's registration message, then he does deregistration, then replaces the original registration message routing information by his information, finally sends faked registration message to the SIP server, along with the same original authentication information. As a result, all calls that be are intended for the target will be directed to the attacker's device.

F- RE-INVETE attack
The goal of method is to modify parameters of established session. The modification can involve changing addresses or ports, adding a media stream, deleting a media stream, and so on. Therefore the attacker can launch faked RE-INVITE message to enforce any unauthorized modification.

5.1.2. Message flooding attacks

This attack involves transmitting a large quantity of forged SIP messages (legitimate or illegitimate) to a targeted SIP-multimedia system. The easiest way to launch these attacks on a SIP proxy server is to flood it with a large number of unwanted SIP requests. As a result, its resources: memory, CPU and bandwidth are exhausted and it is unable to provide service even to the legitimate users [23]. A large quantity of unwanted SIP messages will require the allocation of computational resources for decoding and interpreting. Also the system is busy in treating the faked messages, even the valid messages will be treated at a much slower rate and the overall performance of SIP-multimedia system will decay [24]. With this attack, the system is overloaded with a high amount of the processing and computation of requests that are generated by the attacker, the system will become unavailable for requests from other

users. There are several types of SIP message flooding attack [25]. The most commons are: Register flooding attack, authentication flooding attack, and invite flooding attack.

A- Authentication flooding attack
In this attack, the attacker can generate large number of requests, and respond to each challenge with randomized response. The attacker does not need to calculate the MD5 checksums, any random response will suffice, also the attacker has not valid password, and so all responses will fail. However, the server still has to check each response before rejecting it. In case an attack of this type, the server will be kept busy checking bogus authentication requests and will have less time to process new requests and to handle existing requests [22].

B- Register flooding attack
All SIP devices send REGISTER requests when they are starting and at intervals thereafter. In networks with a large number of deployed devices, the processing load imposes on the servers easily to reach to a point where the application server is too busy in processing REGISTER requests to handle new requests. Malicious REGISTER flooding are abusive problem, the attacker can construct faked REGISTER requests and flood the application server with these requests, where multiple copies of the same spoofed REGISTER request can flood the server.

C- Invite flooding attack
The INVITE flooding attack is similar to the REGISTER flooding. Only the Invite method instead of REGISTER method is utilized to launch the INVITE flooding attack.

5.1.3. Malformed message attacks

This kind of attacks relies on sending large number of malformed message to a SIP server. At best, the server's resources are tied up in processing these bogus messages, at worst, the message triggers a failure in the server or leaves it in an unstable state [17].
SIP parser is developed to receive and process only well-formed messages. However, an attacker is quite possible to generate and transmit various types of malformed messages that are intelligently crafted to exploit vulnerabilities in the SIP parser, resulting to DoS or unstable operation [26]. An attacker can, using a malformed packet, overflow the specific buffers, add large number of characters and modify fields in an illegal form. As a result, the server is tricked to reach to an undefined state, which can lead to request processing delays, and a completely denial of service. We also show how an intelligently crafted single malformed message can crash a server [27]. Malformed message attacks are divided into two classes: structure malformed messages and syntax malformed messages [17].

A- Structure malformed messages
Malformed structure messages do not violate the SIP protocol rules, they conform to the RFC's 3261 syntax, but the complicated structure of the message consumes a time for the parser to process. For example, extra long messages, with multiple header fields and of increased length. Longer message depletes processor power and increases network utilization.

B- Syntax Malformed Messages
In this type, the message does not agree with RFC's 3261 syntax. It violates the SIP protocol rules in a way, where that SIP parser is unable to successfully handle the received messages. For instance, during establishing of SIP multimedia session, an attacker instead of sending a well-formed message, he can send syntax malformed various messages to discover a security problem or flaw of the parser. Consider, for example, an attacker who instead of sending the expected well-formed INVITE message he sends the malformed SIP INVITE message. This message is invalid and cannot be generated under the standard SIP protocol syntax. If the

parser cannot handle null messages, it may crash or it will generate null DNS requests forcing the underlying DNS service to consuming time in looking for host unsuccessfully [13].

5.1.4. Distributed DoS (DDoS)

Distributed DoS (DDoS) has been observed on the Internet for some time [28]. A SIP specific DDoS is possible by generating fake requests that contain spoofed fields (source IP address and via header field), both of them identify the target host falsely as sender of the request. By sending such requests to large number of SIP network nodes on the Internet, the receiving nodes will send these requests to the target host (victim). If the target host ignores these invalid replies, these nodes may keep retransmitting packets to the target host, and thus would amplify the traffic flows directed to the target host and interrupt its services.

5.2. Intrusion Detection Algorithms for DoS Attack

Over the past several years, the computer security community has been developing automated tools to analyze computer system audit data for suspicious user behavior. Intrusion Detection System (IDS) is an important security tool that is used as a countermeasure to preserve data integrity and system availability from attacks. Overall intrusion detection involves detection, prevention, and importantly, reaction to the intrusion attempts.

Intrusion Detection Systems (IDS) have become a standard component in security infrastructures as they allow network administrators to detect policy violations. These policy violations range from external attackers trying to gain unauthorized access to insiders abusing their access.

The goal of IDS is to detect malicious traffic. In order to accomplish this, the IDS monitors all incoming and outgoing traffic. There are several approaches in implementation of an IDS. Among those, two are the most popular (anomaly and misuse detection), as follows:

- *Anomaly detection:* This technique is based on the detection of traffic anomalies. The deviation of the monitored traffic from the normal profile is measured. Different algorithms of this technique have been proposed, based on the metrics that are used for measuring the deviation of normal profile.
- *Misuse or signature detection*: This technique looks for patterns and signatures of already known attacks in the network traffic. A constantly updated database is usually used to store the signatures of known attacks. This technique deals with intrusion detection resembles the way that anti-virus software operates [29].

Attack detection and prevention techniques vary according to attack type, therefore, we will present some of intrusion detection algorithms specific for SIP denial of service attacks, as follows:

5.2.1. Spoofed messages attacks detection algorithms

Cross protocol detection and retransmission mechanism are two main detection techniques which are used to detect SIP signaling attacks. In addition to some of proposed methods by interested researchers, As follows:

- *Cross protocol detection technique*: It was presented in [30, 31] to detect some types of SIP signaling attack relying on this fact. It observes the SIP messages to extract the session information, then, it investigates media traffic after observing BYE message. If RTP traffic is observed after BYE message, it could be highly considered that this is BYE attack. Authors in [32] proposed an abstract intrusion detection framework called SCIDIVE for VoIP systems in general. The system is composed of two detection abstractions: Stateful detection and cross protocol detection.
- *Stateful detection method*: It determines the current state of a subject from multiple packets involved in the same session and detects anomaly using a rule matching engine.

- *The retransmission detection scheme*: It was used in [33] to detect deregistration, BYE, and CANCEL attacks. When SIP server receives one of the mentioned attack's messages, the detection algorithm ask the user to retransmit its last message that is sent to the server. If the retransmitted message is identical to the message that the server received it before, it is recognized as normal message. Otherwise, the server knows that the message was sent from an unauthorized user. To do this, the user must store the last SIP message and retransmit it when it is requested from the server.
- *Conflict Based Attack Detection Algorithm (CBADA):* This method is proposed by [34], it is relying on state conflict to detect some of SIP signaling attack (deregistration, BYE, call hijacking attack), and on message conflict to detect other signaling attacks (CANCEL attack). State conflict is resulted from most of SIP signaling attacks. The attacked SIP entity state is conflicted with the other corresponding entity state. Also message conflict is resulted from signaling attacks. Attacking one SIP entity causes the other to receive unexpected and out of order messages. Attacking the server causes message conflict at the user side, and attacking the user causes message conflict at the server side.

5.2.2. Message flooding attacks detection algorithms

There are several detection algorithms are utilized to provide protection against flooding attacks in SIP-based multimedia systems. We will describe three famous anomaly detection algorithms (Adaptive threshold, Cumulative sum and Hellinger distance), and one misuse detection algorithm is called Weighted Sum.

A- Adaptive Threshold algorithm [34]:

Adaptive Threshold algorithm is a straight forward and simple algorithm, which relies on testing whether the average of a given feature in a predefined time window exceeds a particular threshold. If X_n is the value of the feature in the n^{th} time interval, and μ_{n-1} is the estimated average of the feature from measurements prior to n, then the alarm condition is:

If $X_n > (\alpha + 1)\ \mu_{n-1}$ then ALARM signaled at time n. (1)

$\alpha > 0$ is the amplitude factor, it indicates the percentage above the mean value that one considers to be an indication of anomalous behavior. The mean μ_n can be computed using an Exponentially Weighted Moving Average (EWMA) of previous measurements, as follows:

$\mu_n = \beta\ \mu_{n-1} + (1-\beta)\ X_n$ (2)

where β: is the EWMA factor.

Adaptive Threshold algorithm is used to detect the SIP flooding attack by checking the rate of SIP requests.

B- Cumulative Sum algorithm [35]

Cumulative Sum algorithm (CUSUM) belongs to the family of change point detection algorithms that are based on hypothesis testing to find time of switching from normal to abnormal request rate [36]. The choice of Cumulative Sum algorithm is based on its simplicity in computation as well as its generally excellent performance [37]. Cumulative Sum algorithm was developed for independent distributed random variables $\{y_i\}$. According to the approach, there are two hypothesis θ_0 and θ_1, where the first corresponds to the statistical distribution prior to a change and the second to the distribution after a change. The test for signaling a change is based on the log-likelihood ratio S_n.

$S_n = \sum_{i=0}^{n} s_i$ where $s_i = \ln \frac{P_{\theta_1}(yi)}{P_{\theta_0}(yi)}$ (3)

where:

n: is number of samples, y_i : is requests rate at instant i, s_i: is log-likelihood ratio at instant i.

The typical behavior of the log-likelihood ratio S_n includes a negative drift before a change and a positive drift after the change [34]. Therefore, the relevant information for detecting a

change lies in the difference between the value of the log-likelihood ratio and its current minimum value. Hence the alarm condition for the Cumulative Sum algorithm takes the following form:

If $g_n \geq h$ then an alarm is signaled at time n (4)

where:

$g_n = S_n - m_n$ (5)

$m_n = \min_{1 \leq j \leq n} S_j$ (6)

and: h is threshold parameter.

C- Hellinger Distance algorithm [35]

Hellinger Distance algorithm (HD) measures the deviation between probability measures that does not make any assumptions about the distributions themselves. HD is used to detect anomalies in SIP protocol. For example, we can use some of SIP features which are the number of INVITE, 200 OK, and REGISTER packets arrived in a predefined time-window. HD algorithm consists of training and testing phases. In the training phase, the normalized frequencies p_{INVITE}, p_{200OK}, $p_{REGISTER}$ for INVITE, 200OK, and REGISTER respectively are calculated over the training normal dataset. Similarly, the normalized frequencies q_{INVITE}, q_{200OK}, $q_{REGISTER}$ are calculated in the testing phase for each time-window n or interval. The HD between these frequency distributions of two phases is:

$$HD = \left(\sqrt{p_{INVITE}} - \sqrt{q_{INVITE}}\right)^2 + \left(\sqrt{p_{200OK}} - \sqrt{q_{200OK}}\right)^2 + \left(\sqrt{p_{REGISTER}} - \sqrt{q_{REGISTER}}\right)^2 \quad (7)$$

To keep track of the normal attribute behaviors more accurately, authors in [36] use a dynamic threshold for detection. The threshold value is a function of the average of observed HDs and their mean deviation. Such a dynamic setting of threshold makes an attack harder to evade. They employ the stochastic gradient algorithm to compute the dynamic threshold based on the HD observed during the previous training period. Fast estimators for average ν and mean deviation ε given measurement HD, are computed as follow:

$$Err = HD_n - v_{n-1} \quad (8)$$

$$v_n = v_{n-1} + g \times Err \quad (9)$$

$$\varepsilon_n = \varepsilon_{n-1} + h \times (|Err| - \varepsilon_{n-1}) \quad (10)$$

where:

HD_n is the current sample of the HD, v_{n-1} and v_n are the previous and current means of HD, respectively, ε_{n-1} and ε_n represent the previous and current deviations.

During the testing periods, the Threshold (TH) is computed using the mean of HD and the mean deviation as following:

$$TH_n = x * v_n + y * \varepsilon_n \quad (11)$$

These two factors are adjustable parameters, and can be properly tuned during the training period.

D- Weighted Sum algorithm [34]

Weighted Sum (WSUM) is misuse detection algorithm, it depends on a prior knowledge about attacks signature, it seeks for attacks signature in the incoming samples, this algorithm makes using AET to detect the different types of SIP flooding attacks accurately. The algorithm defines an attack parameter called Attack Effective Factor (AEF), and it equals to the inverse of AET.

$$AEF = \frac{1}{AET} \quad (12)$$

The algorithm can calculate the attack effect during Δt seconds, it is Δt*AEF. In other meaning, during Δt seconds, the attacked server is pushed by Δt*AEF value toward compromised state. To keep trace of the attack effect, the Weighted Sum algorithm samples the incoming requests each Δt seconds. For each sample (i) it calculates the average request rate (λ_i), and then allocates the corresponding AET_i and AEF_i, finally it computes the sample

effect ($\Delta t*AEF_i$). At the sample (n), the attack effect can be computed by cumulating the previous samples effects, calculating Cumulative Attack Effect (CAE), given by:

$$CAE_n = \sum_{i=1}^{n} \Delta t * AEF_i \quad (13)$$

CAE_n reflects the server state at the time n Δt seconds, it expresses how much the server is pushed toward compromised state. When the server is in the normal state the CAE equals to zero. As the server is pushed towards the compromised state, the CAE increases, finally when the server is fully compromised the CAE will be equal to one.

5.2.3. Malformed message attacks detection algorithms

The robust parser of SIP is the first line of defense against the malformed message attacks, parser must discard all non well-formed SIP messages. But, developing immune parser makes the parsing process too complicated and more time consuming, so most of SIP parsers are developed to process only well-formed SIP messages [38]. Some of interested researchers submitted their contribution in this field.

The authors in [38] introduce a complete security framework that deals with malformed messages attacks in SIP implementations and aims at improving the availability, reliability and security level of the provided services. The main idea for the development of such a mechanism stems from the SIP syntax. More specifically, any message that does not comply to SIP RFC can be characterized as malicious. Therefore, the detection mechanism for malformed message attacks can be effectively described through specific structures, known as (attack signatures), which consist of two parts based on the SIP syntax. The first part contributes to the identification of the malformed message, it is a general signature that can be applied to any SIP method. The second part specifies additional rules that can be applied to specific SIP methods as determined by the administrator of each SIP domain, according to the security policy of each SIP-based system provider.

6. Conclusion

SIP is expected to be the future multimedia connections protocol. However, SIP is an evolving protocol, which does not have security built in, therefore it is vulnerable to common attacks in Internet as well as additional attacks. In this paper we present scheme for the more common vulnerabilities in SIP, where we proposed simple classification for these vulnerabilities helping in security analysis. Also, we presented the main potential attacks against SIP-based systems, where we focused on denial of service attacks due to its active effect on the availability of service. We presented classification for these attacks, and explained the most important DoS attacks. Finally, we explained the main intrusion detection algorithms that are used to detect these attacks. This paper is considered very useful to researchers in the field of securing SIP-based multimedia system.

6, References

[1] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., Schooler, E.: SIP: Session Initiation Protocol. RFC 3261 (Proposed Standard) (June 2002) Updated by RFCs 3265, 3853, 4320.

[2] Ruishan Zhang, Xinyuan Wang, Xiaohui Yang, Xuxian Jiang, ”Billing Attacks on SIP-Based VoIP Systems”, Proceeding of the fIrst USENIX workshop on offensive Technology, August 06-10,2007.

[3] Ruishan Zhang, Xinyuan Wang, Xiaohui Yang ,and Xuxian Jiang, “Billing attacks on SIP-based VoIP systems”, Proceedings of the first conference on First USENIX Workshop on Offensive Technologies, Boston, August 2007.

[4] Dorgham Sisalem et. al., SNOCER, Low Cost Tools for and High Available VoIP Communication Services, Towards a Secure and Reliable VoIP Infrastructure, 3rd May 2005, pp. 38-39, www.snocer.org/Paper/COOP- 005892-SNOCER-D2-1.pdf

[5] D. Geneiatakis, G. Kambourakis, T. Dagiuklas, C. Lambrinoudakis, S. Gritzalis, A framework for detecting malformed messages in SIP networks, in: Proceedings of 14th IEEE Workshop on Local and Metropolitan Area Networks (LANMAN), Chania, Crete, Greece, September 2005.

[6] M. Collier, "Basic Vulnerability Issues for SIP Security, Research Report, 2005, available at http://download.securelogix.com, Accessed Sep. 2007.

[7] E. Nuwere, M. Varpiola, "The Art of SIP Fuzzing and Vulnerabilities Found in VoIP", Blackhat Briefings, 2005.

[8] C. Wieser, J. Roning, and A. Takanen, "Security analysis and ex-periments for Voice over IP RTP media streams", Procs. of the 8th Intl. Symp. on System and Information Security (SSI'2006).

[9] The Communications Research Network (CRN). "VoIP loophole aids service deniers?" February 2006.

[10] K. Joongman, "VoIP Secure Communication Protocol satisfying Backward Compatibility," Second International Conference on Systems and Networks Communications (ICSNC 2007), pp.43-43, Aug. 2007.

[11] H. Sengar, D. Wijesekera, H. Wang and S. Jajodia, "VoIP Intrusion Detection Through Interacting Protocol State Machines," Proceedings of the 2006 International Conference on Dependable Systems and Networks (DSN'06), pp. 393-402, Jul. 2006.

[12] Low Cost Tools for Secure and Highly Available VoIP Communication Services (SNOCER), "an research project supported within the Sixth Framework Programme of the EU Commission", http://www.snocer.org/

[13] D. Geneiatakis, T. Dagiuklas, C. Lambrinoudakis, G. Kambourakis, and S. Gritzalis, "Novel Protecting Mechanism for SIP-Based Infrastructure against Malformed Message Attacks: Performance Evaluation Study", 5th International Conference on Communication Systems, Networks and Digital Signal Processing, Patras, Greece, July 2006.

[14] D. Geneiatakis, G. Kambourakis, T. Dagiuklas, C. Lambrinoudakis, and S. Gritzalis, "SIP message tampering: The SQL code injection attack", In Proceeding IEEE of SoftCOM, September 2005.

[15] CERT-In Advisory CIAD-2003-09, Buffer Overrun in RPC Interface Could Allow Code Execution and Denial of Service, August 2003.

[16] J. Fontana, Exchange Server 5.5 Bug Could Be Exploited for Attacks, November 2000. <http://www.pcworld.com/resource/article/0,aid,33882,00.asp>.

[17] Al-Allouni H., Rohiem A., Abd El-Aziz M. H., and El-moghazy A.," VoIP Denial of Service Attacks Classification and Implementation", Proceedings of 26th national radio science conference", Future University, Egypt, March, 2009.

[18] A. Bremler-Barr, R. Halachmi-Bekel, and J. Kangasharju, "Unregister Attacks in SIP", 2nd IEEE Workshop on Secure Network Protocols, 2006.

[19] Amarandei-Stavila Mihai, "Voice over IP Security: A layered approach", XMCO Partners, 2005.

[20] D. Sisalem, and S. Ehlert et al, "General Reliability and Security Framework for VoIP Infrastructures", Technical Report SNOCER, Sep 2005.

[21] R. Sparks, "The Session Initiation Protocol Refer Method", RFC 3515, IETF Network Working Group. April 2003.

[22] Peter Cox, “Reviewing the VoIP Threat Landscape”, Proceeding of The 19th Annual FIRST Conference, SPAIN, June 2007.
[23] E. Nahum et al., “Evaluating SIP Proxy Server Performance,” in Proc. NOSSDAV ’07, 2007.
[24] Hemant Sengar, Haining Wang. Duminda Wijesekera, and Sushil Jajodia, "Detecting VoIP floods using the Hellinger distance," IEEE Trans. on Parallel and Distributed Systems, Volume 19, No. 6, pp. 794-805, June 2008.
[25] T. Magedanz, M. Sher, and S. Wu, “Security Threats and Solutions for Application Server of IP Multimedia Subsystem (IMS-AS)”, IEEE/IST MonAM2006 - Workshop on Monitoring, Attack Detection and Mitigation, Germany, September 2006.
[26] D. Geneiatakis, G. Kambourakis, C. Lambrinoudakis, A. Dagiouklas, and S. Gritzalis, "A framework for protecting SIP-based infrastructure against Malformed Message Attacks", Science Direct - Computer Networks, Volume 3, No. 10, pp. 2100-2113, Elsevier, 2007.
[27] T. J. Walsh and D. R. Kuhn,”Challenges in Securing Voice over IP”, In IEEE Security& Privacy, June 2005.
[28] Rescorla, E. (2000), SSL and TLS Designing and Building Secure Systems. Addison Wesley.
[29] Mithcell Rowton, Introduction to Network Security Intrusion Detection, December 2005.
[30] Y. Wu, S. Bagchi, S. Garg, and N. Singh, “SCIDIVE: A Stateful and Cross Protocol Intrusion Detection Architecture for Voice-over-IP Environments”, Proceedings of the International Conference on Dependable Systems and Networks, p 433 – 442, July 2004.
[31] H. Sengar, D. Wijesekera, H. Wang, and S. Jajodia, “VoIP Intrusion Detection Through Interacting Protocol State Machines, In Proceedgins of the 2006 International Conference on Dependable Systems and Networks (DSN 2006), June 2006.
[32] Y. Wu, S. Bagchi, S. Garg, N. Singh and T. Tsai, “SCIDIVE: A Stateful and Cross Protocol Intrusion Detection Architecture for Voice-over-IP Environments”, 2004 International Conference on Dependable Systems and Networks (DSN'04), Florence, Italy, 2004.
[33] Cha, H. et al, "Detection of SIP De-Registration and Call-Disruption Attacks Using a Retransmission Mechanism and a Countermeasure Scheme", IEEE International Conference on Signal Image Technology and Internet Based Systems, p 650, 2008.
[34] Husam Al-Alouni, “security of voice over internet protocol”, PhD of science thesis, military technical college, Cairo, 2010.
[35] M. Akbar, Z. Tariq and M. Farooq,” A Comparative Study of Anomaly Detection Algorithms for Detection of SIP Flooding in IMS”, In 2nd International Conference on Internet Multimedia Services Architecture and Applications, India, 2008.
[36] M. Basseville and I. V. Nikiforov,” handbook of Detection of Abrupt Changes: Theory and Applications”, Prentice-Hall, 1993.
[37] H. Wang, D. Zhang, and K. Shin,”Detecting SYN flooding attacks”, in Proceedings of Annual Joint Conference of the IEEE Computer and communications Societies, February, 2002.
[38] D. Geneiatakis, G. Kambourakis, C. Lambrinoudakis, A. Dagiouklas, and S. Gritzalis, "A framework for protecting SIP-based infrastructure against Malformed Message Attacks", Science Direct - Computer Networks, Volume 3, No. 10, pp. 2100-2113, Elsevier, 2007.