COMPREHENSIVE COMPARISON OF VOIP SIP PROTOCOL PROBLEMS AND CISCO VOIP SYSTEM

Dr TALAL AL-KHAROBI[1] and Mohmmed Abduallah Al-Mehdhar[1]

[1]Department of Computer Engineering, King Fahd University of Petroleum & Minerals (KFUPM), Dhahran, Saudi Arabia
Talalkh@kfupm.edu.sa
[1]Department of Computer Engineering, King Fahd University of Petroleum & Minerals(KFUPM), Dhahran, Saudi Arabia
G200804340@kfupm.edu.sa

ABSTRACT

Voice over IP (VoIP), use of the packet switched internet for telephony, has improved substantially in the past few years. On the other hand, VoIP has many challenges that do not exist in the public switched telephone network (PSTN), a circuit switched system. VoIP is an application running on the internet, and therefore inherits the internet's security issues. It is important to realise that VoIP is a relatively young technology, and with any new technology, security typically improves with maturity. This paper provides a comprehensive comparison of a VoIP SIP protocol and CISCO VoIP system. The comparison involves the investigation of the vulnerabilities that target both systems and how secure each system is. With this comparison we present our conclusion on which system is more secure.

KEYWORDS

SIP, SKINNY, DOS, Cisco, and CCSCP

1. INTRODUCTION

Voice-over-IP (VoIP) implementations enable users to carry voice traffic over an IP network. The main reasons for the evolution of the Voice over IP market are low cost phone calls, add-on services and unified messaging and merging of data/voice infrastructures [3]. A VoIP system consists of a number of different components such as Gateway/Media Gateway, Gatekeeper, Call agent, Media Gateway Controller, Signalling Gateway and a Call manager [4]. The Gateway converts media provided in one type of network to the format required for another type of network [4]. For example, a Gateway could terminate bearer channels from a switched circuit network and media streams from a packet network (e.g. RTP streams in an IP network)[3]. The gateway may be able to process audio, video and T.120 alone or in any combination, and is capable of full duplex media translations. The Gateway may also play audio/video messages and perform other IVR functions, or may perform media conferencing. In VoIP, the digital signal processor (DSP) segments the voice signal into frames and stores them in voice packets. These voice packets are transported using IP in compliance with one of the specifications for transmitting multimedia (voice, video, fax and data) across a network: H.323 (ITU), MGCP (level 3, Bellcore, Cisco, and Nortel), MEGACO/H.GCP (IETF), SIP (IETF), T.38 (ITU), SIGTRAN (IETF), Skinny (Cisco) etc. Coders are used for efficient bandwidth utilisation [3]. The coder decoder compression schemes (CODECs) are added for both nodes of the connection and the conversation proceeds using Real-Time Transport Protocol /User Datagram Protocol/Internet Protocol (RTP/UDP/IP) as the protocol stack. Quality of Service , a number of high level ways are used to overcome the oppose environment of the IP network and to provide a good Quality of

DOI: 10.5121/ijnsa.2012.4409

Service [6]. As VoIP is very sensitive to delay (delayed – sensitive), a well-engineered, end-to end network is necessary to use VoIP successfully [6]. There are several methods and algorithms developed to evaluate the QoS: PSQM (ITU P.861), PAMS (BT) and PESQ. Each offers a specific level of QoS. The quality of transmitted speech is a subjective response of the listener. A common measurement used to determine the quality of sound produced by specific CODECs is the mean opinion score (MOS) [6]. With MOS, a wide range of listeners judge the quality of a voice sample (corresponding to a particular CODEC) on a scale of 1 (bad) to 5 (excellent).

Services:

The following are examples of services provided by a Voice over IP network according to the market.

Requirements:

Phone to phone, PC to phone, phone to PC, fax to e-mail, e-mail to fax, fax to fax, voice to e mail, IP Phone, transparent CCS (TCCS), toll free number (1-800), class services, call centre applications, VPN, Unified Messaging, Wireless Connectivity, IN Applications using SS7, IP PABX and soft switch implementations [4].

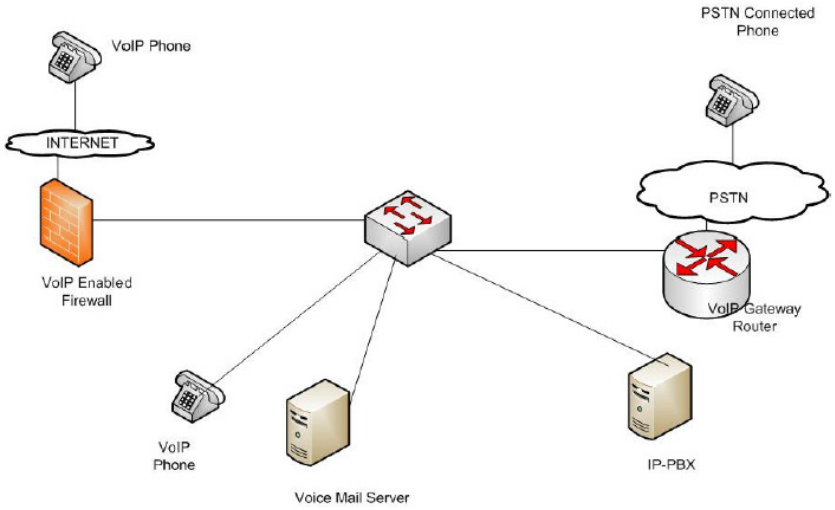

Figure 1: Typical Network structure

2. VOIP ARCHITECTURES

When using IP protocol, three different types of connections can be used to set up a call: (1) PC to PC, where nodes talk online using their PCs; (2) PC to telephone, where nodes make and receive voice calls and messages while on the Internet; and (3) telephone to telephone, where calls are made and received using phones connected to the Public Switched Telephone Network (PSTN) or IP telephones connected to a data net [3]. VoIP uses the Real-Time Protocol (RTP) for transport, the Real-Time Transport Protocol (RTTP) for reporting Quality of Service (QoS), and H.323, SIP, MGCP (Media Gateway Control Protocol/Megaco) for signalling [8]. These protocols operate in the application layer; that is, on top of the IP protocol. Most current VoIP implementations use the H.323 protocol, the same protocol used for IP video. Below are the UML models for the architectures implied by these standards [4].

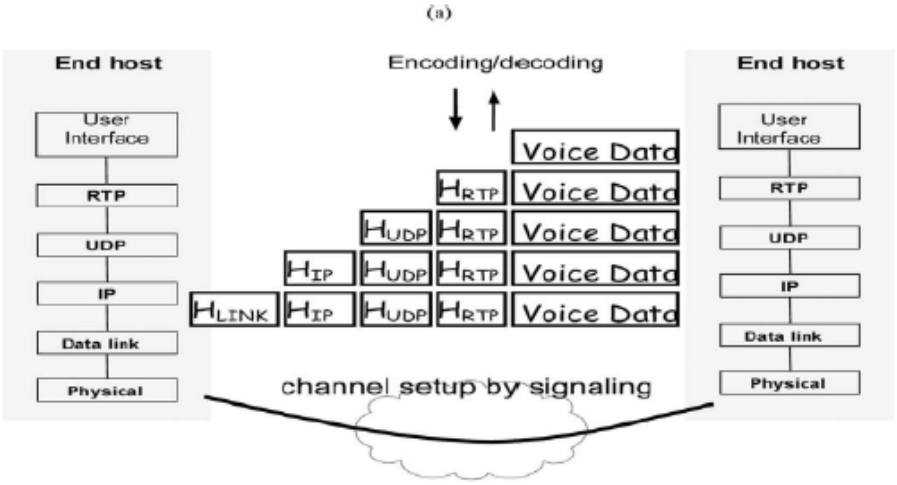

Figure 2: Data processing Structure

2.1 VoIP data processing

The VoIP data processing consists of the following four steps: signalling, encoding, transport, and gateway control [1].

- **Signalling:** The main purpose of the signalling protocol is to create and manage connections or calls between endpoints. H.323 and the session initiation protocol (SIP) are two widely used signalling standards for call setup and management.

- **Encoding and Transport:** Once a connection is set up, the voice must be transmitted by converting it into digitised form, then segmenting the voice signal into a stream of packets. The first step in this process is converting analogue voice signals to digital, using an analogue-to digital converter. Here a compression algorithm can be used to reduce the volume of data to be transmitted. Next, voice samples are inserted into data packets to be carried on the Internet using typically the real-time transport protocol (RTP)[3]. RTP packets have header fields that hold the data needed to correctly reassemble the packets into a voice signal at the other end. Lastly, the encapsulated voice packets are carried as payload by the user datagram protocol (UDP) for ordinary data transmission. At the other end, the process is reversed: the packets are disassembled and put into the proper order, and then the digitised voice is processed by a digital-to-analogue converter to render it into analogue signals for the called party's handset speaker. Fig. 1 illustrates the basic flow of voice data in a VoIP system [2].

- **Gateway Control:** The IP network itself must then ensure that the real-time conversation is transported across the telephony system to be converted by a gateway to another format—either for interoperation with a different IP-based multimedia scheme or because the call is being placed onto the PSTN. With the switch to the Internet as a carrier for voice traffic, we see some of the same security issues that are prevalent in the circuit switched telephone network, such as eavesdropping and toll fraud. We are also exposed to new types of attacks that are more prevalent in the data world of the Internet, such as denial-of-service (DoS) attacks [7].

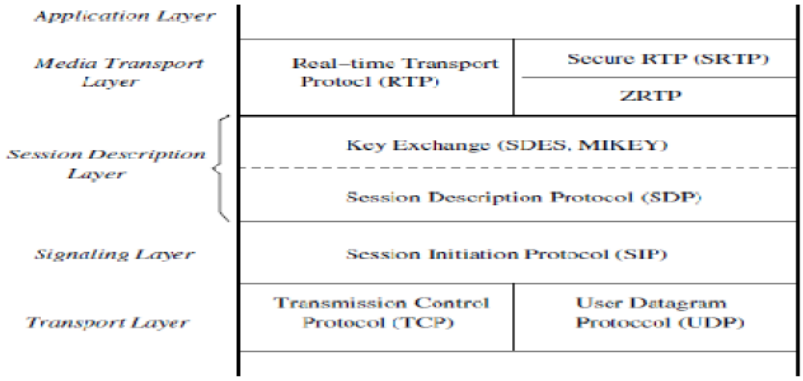

Figure 3: SIP Stack Architecture

3. SIP PROTOCOL

SIP [11] is an application layer protocol used for establishing and tearing down multimedia sessions, both unicast and multicast. It has been standardised within the IETF for the invitation to multicast conferences and VoIP services. The SIP user agent has two basic functions:

- Listening to the incoming SIP messages
- Sending SIP messages upon user actions or signalling protocol used for creating, modifying and terminating sessions with one or more nodes. User Agents (UA) represent phone devices or software modems. SIP users are not bound to specific devices: nodes register with the registrar and use an address in a special form to identify other users [6]. SIP URI special type of Uniform Resource Identifier (URI) to identify SIP users, similar to email addresses. A location server stores the address bindings of users when they register themselves with the registrar. Proxy mode or Redirect mode are SIP server use one of them. In the proxy mode, the server intercepts messages from the end points, and will inspects : field, contacts the location server to get the username into an address and send the message to the end point or another server. Forking proxies receive a single request and send it to multiple recipients (this makes SIP potentially vulnerable to denial of service attacks). In the redirect mode the only difference is that, instead of forwarding the packet, the redirected server returns the address to the end points and the responsibility for transmitting packets is put on the end points [2].

SIP uses a HTTP-like request-response mechanism for initiating a two-way communication session. The protocol itself is modelled on the three-way TCP handshake. In order to set up a connection between Alice's and Bob's UAs, Alice's SIP URI is first resolved into the IP address of the UA under which Alice is currently registered. SIP address resolution and routing is usually not done by the UA itself, but rather delegated to the proxy server for the UA's domain. In our example, Bob's proxy will make a DNS lookup to determine the address of Alice's proxy server. During the setup process, communication details are negotiated between UAs using the Session Description Protocol (SDP). To start a call to Alice, Bob's UA sends an INVITE request to the proxy server containing SDP, which is then sent to Alice's UA. If Alice accepts Bob's call, she sends an OK message back to Bob containing her SDP. Bob then responds with an ACK. Media exchange takes place directly between Alice's and Bob's respective UAs.

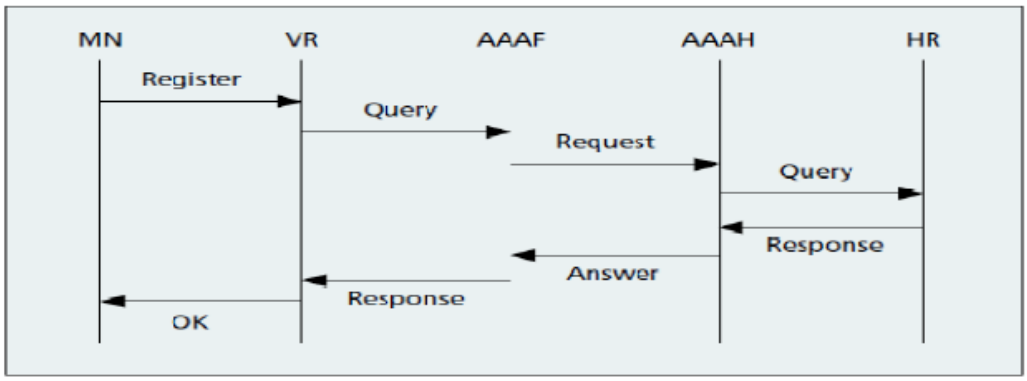

Figure 4: SIP Message Flow

3.1. SIP Message Flow

We assume that the MN and foreign network use Dynamic Host Configuration Protocol (DHCP) or one of its variants to configure its sub network. The MN broadcasts DHCP_DISCOVER message to the DHCP servers. Several servers may offer a new address to the MN via DHCP_OFFER that contains an IP address, the address of a default gateway, subnet mask, and so on. (There is a proposal that DHCP_OFFER can also include SIP information [13]). The MN then selects one DHCP server (and an IP address) and sends DHCP_REQUEST to the selected server. The DHCP server sends DHCP_ACK to confirm the assignment of the address to the MN. After the MN is assigned an IP address from the DHCP server, the MN will initiate the signalling flow for SIP complete registration in a visited network, as depicted in Fig. 4 [10]. (DHCP message exchange is not shown here.) First, the MN sends a SIP REGISTER message with its new (temporary) IP and MN's profile to the VR. Note that the MN has obtained the address of the local SIP proxy server from DHCP messages upon its configuration (or reconfiguration) in the visited network. The VR queries the AAA entity of the visited network to verify the MN's credentials and rights by sending a Diameter-compliant message (QUERY in Fig. 4). The AAA entity (AAAF) of the visited network sends a request (Diameter compliant message) to the AAA entity (AAAH) of the home network to verify the MN's credentials and rights. The AAAH queries the HR and gets a reply from the HR, and then sends the appropriate answer to the AAAF. The AAAF sends an appropriate response to the VR. The VR sends either an SIP 200 OK response to the MN upon success, or a 401 unauthorised response upon failure of the registration. Note that the messages to/from AAA servers are Diameter compliant. After this registration, the MN can initiate the SIP session by sending the INVITE message to the caller. (Suppose the MN is the caller and a correspondent node, CN, is the caller.) Then the caller responds with a SIP OK message. (These messages are not shown in Fig. 4.) Here, we assume that the CN is located in its home network. For a detailed description of the signalling messages in SIP, please refer to [11].

In the case of micro mobility, there is no need to verify the user's credentials via the AAA server. The MN (SIP client) sends a SIP REGISTER message with the new MN's address. Then the VR verifies the user's credentials and registers the user of the MN in its contact database, and updates its contact list, which is called expedited registration. And then the VR replies with a SIP OK message. In the case of macro mobility, the signalling message flow is the same as the SIP registration (Fig. 4).

4. SKINNY CLIENT CONTROL PROTOCOL (SCCP)

Skinny Client Control Protocol (SCCP) is a Cisco proprietary protocol used between Cisco Call Manager and Cisco VOIP phones. It is also supported by some other vendors. However, Skinny International Journal of Network Security & Its Applications (IJNSA), Vol.4, No.4, July 2012 142 does not mean low features or functions while SCCP in VoIP that Cisco uses with its 'fat' telephone equipment systems. Skinny reduces the processing load on its hardware.

4.1. Protocol Structure - SCCP (Skinny)

Skinny Client Control Protocol: The skinny client (i.e. an Ethernet Phone) uses TCP/IP to transmit and receive calls and RTP/UDP/IP to/from a Skinny Client or H.323 terminal for audio. Skinny messages are carried above TCP and use port 2000.

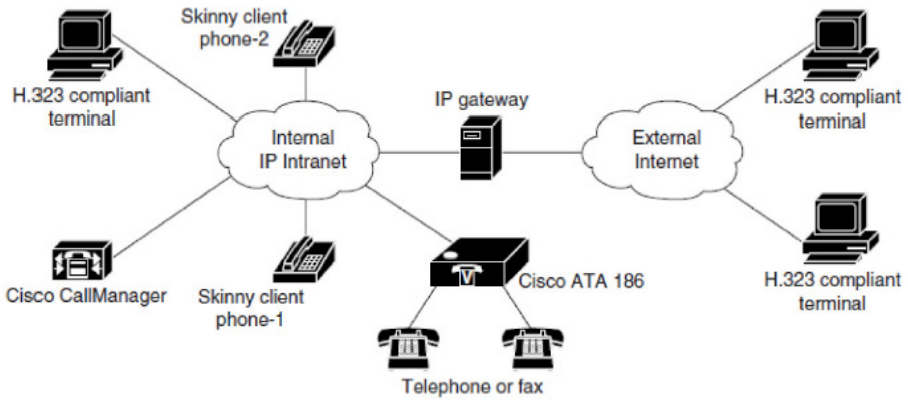

Figure (5) Skinny Protocol Architecture

4.2. Cisco Unified Call Manager (CUCM):

Cisco Unified Call Manager (CUCM) is the call processing component of the Cisco IP telephony solution which extends enterprise telephony features and functions to packet telephony network devices such as IP phones, media processing devices, voice-over-IP (VoIP) gateways, and multimedia applications.

4.3. Cisco Unified Presence Server (CUPS):

Cisco Unified Presence Server (CUPS) is the identity tracking component of the Cisco IP telephony solution which collects information about a user's availability status, such as whether or not you are using a communications device such as a phone at a particular time. It also collects information regarding a user's communications capabilities, such as whether Web collaboration or video conferencing is enabled.

4.4. Skinny Call Control Protocol (SCCP)

Skinny Call Control Protocol (SCCP) is a Cisco proprietary voice protocol used to facilitate call management functions between Call Manager systems and IP phones. SCCP uses TCP port 2000 for communications and Secure SCCP (SCCPS) running on TCP port 2443.

4.5. How it Works

Cisco allows skinny clients to communicate with Call Manager over TCP/IP using a messaging set called Client Control Protocol (SCCP). Skinny messages are transferred via TCP and use port 2000. Skinny gateways are a series of digital gateways that include the DT-24+, the DT- 30+, and the WS-X6608-x1 Catalyst voice module. SKINNY systems use a proxy for the H.225 and H.245 signalling, and use RTP/UDP/IP for audio. The skinny client uses TCP/IP to transmit and receive calls and RTP/UDP/IP to/from a Skinny Client or H.323 terminal for audio. The end user of a LAN or IP- based PBX must be simple to use, friendly and cheap. While the H.323 is quite an expensive system, skinny allows skinny clients to communicate with H.323 proxy using the SCCP. A proxy is used for the H.225 and H.245 signalling. When two skinny clients communicate they use RTP over UDP, but when calling a non-skinny client the clients establish a connection through the Call Manager using TCP and then the two endpoints communicate using UDP. Moreover, SCCP also uses Transport Layer Security (TLS) to encrypt communications and provide for the confidentiality of voice conversations.

5. MY CONTRIBUTIONS

I am analysing the security of VoIP protocols mainly in SIP protocol, which is the main protocol in VoIP these days. Moreover, I am suggesting a solution for some problems regarding SIP and common security problems.

6. ATTACKS AND VULNERABILITIES

6.1 Attacks and Vulnerabilities in SIP :

SIP deployment in an IP network is exposed to a large variety of different threats, for example ID and Internet. ID: Displaying the right ID of a caller is a legal requirement for phone companies. What happens if someone fakes their ID? The main reason why the Internet is not safe is that there have never been enough safeguards and equipment to keep a network totally safe on the Internet. A SIP-based network will face two different threats. These are internal and external threats. The external threats are attacks launched by an aggressor who is not participating in the actual SIP-based communication. The external threats arise when the information crosses boundaries/networks which involve a third-party or untrustworthy networks. The other threat is the internal threat. This is often a threat launched by an SIPsession participant. Because an SIP session participant is launching the attack the participant can no longer be trusted. Because the network is protected by firewalls and so no one expects attacks from the inside, these attacks are more complex and it is much more difficult to find the source of the attack.

6.1.1. Denial of Service attacks:

This kind of attack works by sending lot of strange, malformed or other types of packets to a server or gateway, and then the huge traffic is redirected to the victim node to make this stop responding. SIP architecture that gives the attacker the advantage of using this kind of attack, by using a spoofing router header request and but the IP address of the targeted node in this request and sending it to forking proxies, so the proxies will send a huge number of reflected messages to that node [5]. Another approach could be used in the same type of attacks called Reflection in which the attacker can send a spoofed request to many nodes and proxies and but the victim's address in the header so each one of those nodes and proxies will replay and overwhelm that node. The Domain Name Service has the main role in all SIP networks with the three following factors [15].

1. Qualified Domain Names (FQDN) that need further processing from SIP entity.

2. Allows the mapping of a PSTN telephone number to SIP number, when this mapping has been done previously using the domain name service to make interconnection between the Public Switched Telephone Network (PSTN) and an SIP network.

3. SIP entity issues a DNS SRV [17] request for the domain regarding SIP URI to find its right contact server. This kind of attack could be done at any kind of SIP entity (user agent, proxy, registrar, and redirect and mostly it targeted proxies or registrars/redirectors [2]. When an SIP server meets a fully qualified URI in a header field required for routing, it sends a query to the name server to receive address mapping. 1.3 DNS queries in SIP encounter less than 100ms until they receive an answer, but unfortunately SIP can suffer from considerably higher latency due to configuration errors [19]. Disturbing an SIP server with a high processing request is the main goal in DNS attack, while using a special SIP request containing URI which mostly is not cached in DNS server and requires a high processing time, where these uncached URI will be directed to an authoritative name server which actually has a low response time. The DNS attack is easy to generate by modifying the SIP packet header, by adding random host names to the left side of the address domain. The latter case can be easily discovered by querying different name servers and measuring reply times. As an example of an SIP message that meets with the SIP standard and such messages cannot be easily detected by an Intrusion Detection System or filtered out by a SIP server [7].

Establishing SIP queries with a different of URIs (SIP Message with Unresolvable URIs) will stop operation at a SIP server for a respectable time, so the SIP server can only continue its operation after having received an answer from the DNS server. For example, the SIP server will wait up to five seconds from a BIND DNS server [20] which is commonly used to resolve a request. If it does not receive any answer from the BIND DNS server within five seconds, this domain name will be marked as unresolvable and the SIP server will continue to deal with the next one.

Figure 6: Ethereal, graphic analysis of a VoIP call

6.1.2. Eavesdropping:

This type happens when the attacker have access data transmit between both nodes .Moreover these data could be used to replay an attack or for illegal usage and not only voice data could be used but also all the data services offered by VoIP, such as Fax, Password exchange during sessions and maybe dual-tone multi-frequency (DTMF) to get a bank or credit card password in voice banking services [8]. It is possible to get an MITM attack in a wired network via known approaches.

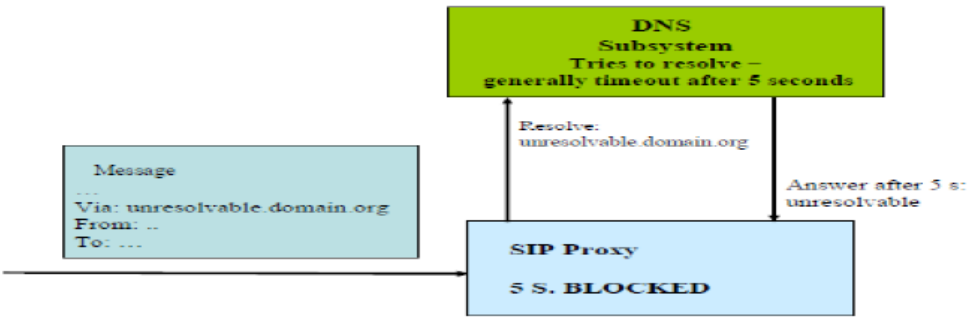

Figure 7: The Attacking Scenario by blocking SIP proxy with messages containing unresolvable URIs.

using an ARP harmful attack to force the SIP proxy, and the VoIP telephones to join a communication with a malicious third party [4]. The Ethereal sniffer offers functions to recognize the VoIP calls in the sniffed packets, using the SIP protocol, and to resemble the audio stream starting from the payload of the captured RTP packets. Ethereal is able to rebuild graphically all the exchanges between the two communication parties in the selected call (Figure 6). Figure 6: Ethereal, graphic analysis of a VoIP call. Furthermore, Ethereal is able to recognise the various RTP streams in the captured packets. Figure (6) Analysis VoIP call using Ethereal.

6.1.3. Authentication:

Authentication provides a mechanism to verify the legitimacy of a user or a client. In an SIP network, the authentication can be placed between the user agent and the proxy, where the proxy server requires a user agent to authenticate itself before processing an "invite" message from it. In the same way, a user agent can request authentication of a proxy or redirect server.

6.1.4. Cancel Attack

SIP uses INVITE message to initiate a call, however the INVITE message is not usually encrypted and attackers could modify fields necessary to recreate the forged SIP CANCEL message for the sniffing SIP INVITE Packets. Moreover, we cannot differentiate between the normal SIP CANCEL message and the faked one, because the faked CANCEL packet includes the normal fields that come from the SIP INVITE message. The goal of the SIP CANCEL attack is to prevent the normal call from begin initiate. When a victim is waiting for calls, as soon as the attacker catches a call invitation message for a targeted node, it will send an SIP CANCEL message (faked message) using sniffed a INVITE SIP message, which makes the call establishment fail.

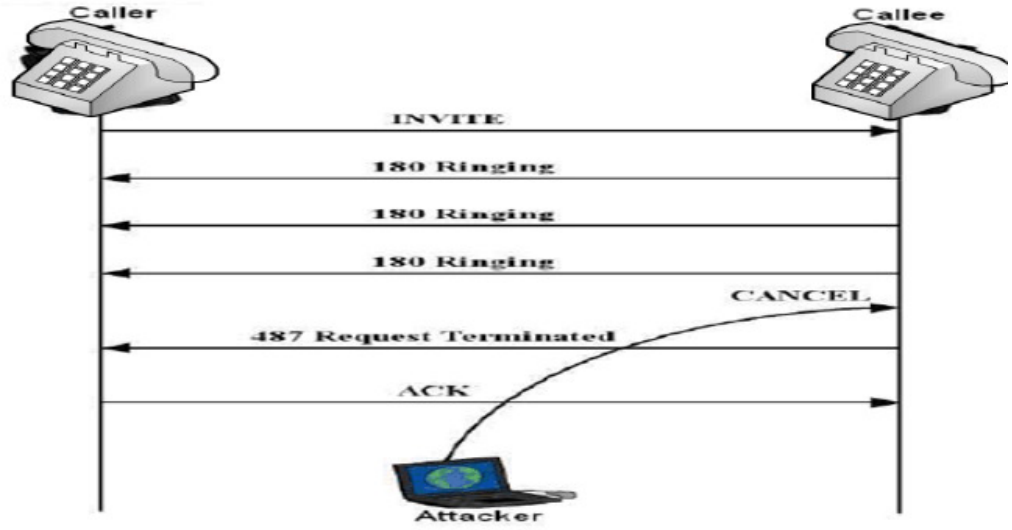

Figure 8: Cancel Attack

6.1.5. BYE Attack

BYE DoS Anomaly Traffic Detection SIP uses the BYE messages field as the same authentication field is in the SIP INVITE message for security and accounting issues. However, attackers can recreate BYE packets by sniffing SIP INVITE packets. The faked SIP BYE message is the same as the normal SIP BYE. The attacker can terminate a normal call between two nodes using a sniffed SIP INVITE message, then generating a BYE message to the SIP proxy server. In the SIP BYE attack, it is not easy to differentiate between normal call termination procedure and BYE attack.

Man-in-the-Middle Attack

This kind of attack works when the attacker has access to both nodes, and the attacker can change the content of the conversion by Play-back of another sniffed call. It works as follows: when Alice is inviting Bob, Eve redirects the request to her phone. Her response to the call to Alice and at the same time sends another invite to Bob. Eve uses a Diffie-Hellman key exchange with Alice and Bob and establishes two different keys for encrypting the media stream. Alice thinks she is talking to Bob and Bob thinks that he is talking to Alice, but they are talking to Eve, where Eve has the ability to change the conversation. When anyone requests an SAS confirmation, Eve also relays the SAS and confirms the two different Sass, which grants to two spate keys for Alice and Bob. This type of attack will succeed only if the Diffie-Hellman public values are not authenticated.

6.1.6. RTP Payload / Tampering attack

RTP is a UDP extension which adds sequencing information. The VoIP RTP protocol used to carry encoded voice messages between two communication nodes. In the man-in-the-middle attack, to get access to the RTP message transfer between two nodes, an attacker can listen or modify the payload of the message. In this case if an attacker can modify the payload of messages, they can either add noise or put their own information into the packet. This will jam or make impossible conversation between the nodes on the call. The RTP Tampering is the same as

an eavesdropping attack, where it happened If the attacker resequanced or made the packets not useful by modifying the sequence number and timestamp fields in the RTP packet header.

This attack can either make the conversation unclear, or in some protocol stacks, it cuts off the node receiving the packets, which switches the node offline until the software is restarted [7].

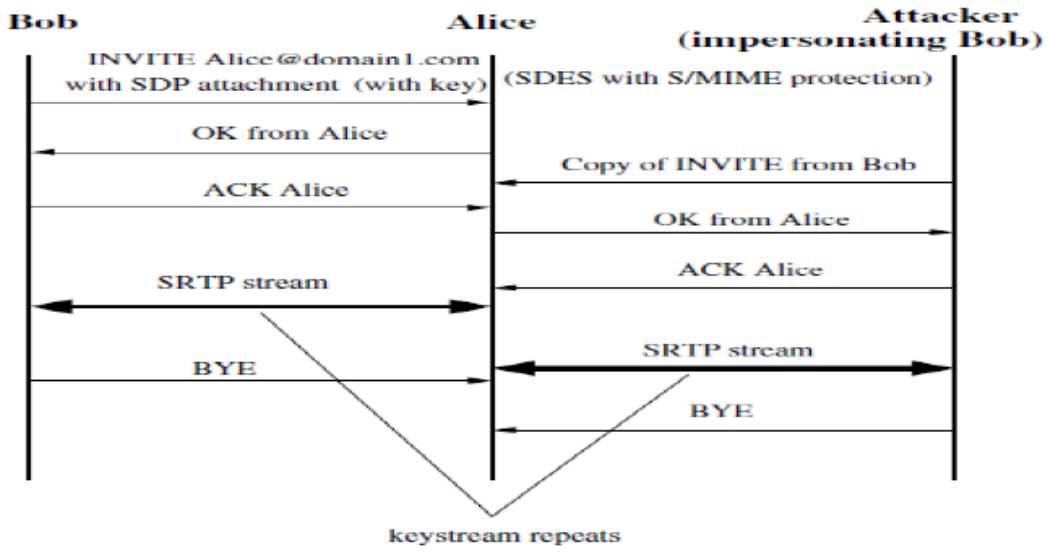

Figure 9: RTP attack Process Flow

6.1.7. Attack on SDES/SRTP

Figure 9 shows an attack on SRTP when used in combination with SDES key exchange. When Alice and Bob have completed their call using VoIP , but there were attacker who is passively eavesdropping , without knowing the session key, so he is not able to decrypt the streamed data. We assume Bob who initiates the call, and SRTP uses SDES transport key material. To provide confidentiality for the SDES message, S/MIME was used to encrypt the payload. S/MIME, in general, is preferred over TLS for protecting SDP messages because (I) S/MIME provides end-to end integrity and confidentiality protection, and (ii) S/MIME does not require the intermediate proxies to be trusted. S/MIME does not provide any anti-replay protection. After the original call has been finished, the attacker can start a new call (replay) using Bob's INVITE message to Alice. This INVITE message will have an S/MIME-encrypted SDP attachment with the SDES key transfer message. (Fig. 4 shows the sessions running concurrently, but the attack need not be adaptive; one session can be executed after the other.) However, Alice does not maintain state info for SDP, so Alice will not be able to discover if it is a new call or just a replay. Using the session key for the old call as Eve's HMAC key, she will get the same master and Salt in the first call. The SSRC and sequence number are the same as the old call, and the keystream created by using AES to the (key, SSRC, SEQ) triple will be the same as in the original previous call. SRTP encryption simply is xor of the data stream with the keystream. Then if Alice sends a datagram in the new call that she thinks she is calling Bob, the attacker can xor the encrypted data stream with the data stream he collected in the previous call. The keystream will cancel out, and the result will be the xor of two data streams. The amount of redundancy and the ability of guessing for the attacker will specify if the attacker will able to resemble partially or completely the stream of both calls.

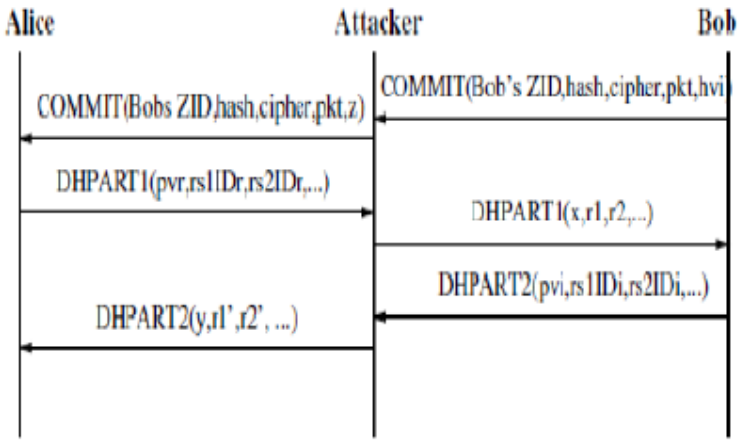

Figure 10: Man-in-the-Middle Attack

6.1.8. ZRTP Attack

ZRTP suffers from two types of attack. The first one is Denial of Service (which is described above). ZRTP is vulnerable to denial of service attacks. This type of attack is caused when the attackers send fake HELLO messages to the end victim. ZRTP node (the victim) will create a half-open connection in response to each HELLO message, and keep their parameters in memory. After a long period, the victim will run out of memory, and requests from legitimate clients will be dropped.

6.1.9. Authentication

This is the second vulnerability in ZRTP. The main advantage of ZRTP is that it has no need for the global trust associated with a public key Infrastructure. ZRTP idea depends on a Short Authentication String (SAS), which is a (keyed) cryptographic hash of Diffie-Hellman values with pre-shared secrets. At the end of each session, nodes using update and keep shared secrets for the next session of authentication [7].

6.1.10. ZRTP Man-in-the-Middle

This attack happens when two nodes that have already had several sessions, the attacker must be present from the beginning and on each session. Each ZRTP user keeps shared secrets rs1 and rs2 for users with whom he previously communicated. When starting a new session, the user sends his ZID, which is used by the recipient to resolve the set of shared secrets linked with ZID. The attacker can compute the session key now by hashing the Diffie-Hellman value concatenated with the shared secrets. However, if the attacker compromises the Diffie-Hellman value, he cannot compute the session key because he does not know the shared secrets (shared secrets are re-computed after each session), the attacker must sniff every session. So ZIDs is the main problem with the ZRTP, since ZIDs are used by recipients to retrieve shared secrets, and are not authenticated early in the protocol exchange. The attacker sniffs a session between Alice and Bob and learns Bob's ZID. He then starts a Man-in-the-Middle attack as shown in Figure (?).x = gx' mod p and y = gy' mod p are two random exponents x', y' chosen by the attacker.

Then the attacker hash x concatenated with the set of algorithms chosen by Bob for the ZRTP session to get z. The attacker also changes all shared-secret IDs with random numbers. When Alice receives the DHPART2 message from Bob, she retrieves the set of secrets that she shares with Bob and computes the set of IDs, so they do not match because the attacker has changed all the IDs. However, ZRTP specification allows the set of shared secrets to be empty: "the final shared secret, s0, is calculated by hashing the concatenation of the Diffie-Hellman shared secret (DHSS) followed by the (possibly empty) set of shared secrets that are actually shared between the initiator and responder" [31, p. 12]. The specification does not need Alice to stop the protocol, but it instructs her to compute the joint Diffie-Hellman value, such as sir mod p (=gy'.svr mod p). The session key is now computed as the hash of the joint Diffie-Hellman value alone because Alice believes that she does not have any shared secrets with Bob anymore. Similarly, Bob computes the session key as the hash of the Diffie-Hellman value g'x.svi mod p. The attacker knows both values, so he can compute SRTP master key and salt, and completely break the SRTP encryption [9].

6.2. Attacks and Vulnerabilities in Skinny

Cisco also suffers from some attacks and vulnerabilities that affect the services and the security of users, such as:

6.2.1. Skinny Client Control Protocol (SCCP) firmware buffer-overflow vulnerability:

This attack happens when parsing DNS responses. An attacker can exploit this vulnerability using a specially crafted DNS request using TCP/UDP port 53 to the victim devices. This issue affects Cisco Unified IP Phones 7940, 7940G, 7960, and 7960 running SCCP firmware prior to 8.0(8). This vulnerability is being tracked by CVE-2008-0530 and Cisco Bug IDs CSCsj74818 and CSCsk21863 [19].

6.2.2. Cisco SIP firmware a buffer-overflow vulnerability:

This attack happens when handling Multipurpose Internet Mail Extensions (MIME-encoded) data. An attacker can exploit the vulnerability by sending a specially crafted SIP message to the victim's device using TCP/UDP port 5060. This issue affects Cisco Unified IP Phones 7940, 7940G, 7960, and 7960G running SIP firmware prior to 8.8(0). The vulnerability is being tracked by CVE-2008-0528 and Cisco Bug IDCSCsj74786 [20].

6.2.3. Internal telnet server buffer-overflow vulnerability:

This attack affects the device's running of the SIP firmware. Specifically, the device cannot deal with the specially crafted commands it receives. Attackers can exploit this vulnerability by creating and sending a specially crafted command via TCP port 23. The affected Cisco Unified IP Phones 7940, 7940G, 7960, and 7960G running SIP firmware prior to 8.8(0). This vulnerability is being tracked by CVE-2008-0529 and Cisco Bug ID CSCsj78359 [18].

6.2.4. Heap-based buffer-overflow vulnerability:

This attack targets phones running the SIP firmware. This vulnerability occurs when handling challenge/response messages from a SIP proxy. If an attacker controls the SIP proxy or behaves as a 'Man-in-the-Middle' that the phone is registered to or attempts to register, the attacker can send malicious challenge/response messages. This vulnerability is being tracked by Cisco Bug ID CSCsj74786 [18].

6.2.5. SQL Injection Vulnerabilities:

Cisco Unified Communications Manager is vulnerable by the SQL injection vulnerability. This vulnerability could allow an authenticated, remote attacker to modify the system configuration to create, modify and delete Users or modify the configuration of the Cisco Unified Communications Manager. This vulnerability is documented in Cisco Bug ID CSCtg85647 and CSCtj42064 [20].

6.2.6. Denial of Service Attack:

Cisco could be vulnerable to many types of Denial of Service attacks that will affect different parts of the VoIP systems and lead to system close or bad services:

6.2.6.1. SCCP DoS:

A sequence of specially crafted packets could be sent by a remote user to the SCCP port (2000) and SCCPS port (2443) to cause the target Call Manager to crash. This would negatively affect the voice services. Cisco has assigned Cisco Bug ID CSCsf10805 to this SCCP/SCCPS vulnerability [17].

6.2.6.2. ICMP Echo Request Flood Denial of Service:

Lots of ICMP Echo Requests could be sent by a remote user that targets Call Manager or Cisco Unified Presence Server to cause various services to crash. This will certainly impact on voice services. Cisco has assigned Cisco Bug IDs CSCsf12698 and CSCsg60930 to this ICMP vulnerability for the Call Manager and Presence Server, respectively [17].

6.2.6.3. The IPsec Manager DoS:

The IPsec Manager Service could also be targeted by a remote user using a specially crafted UDP packet to the IPsec Manager service on UDP port 8500 to cause the service to fail. This would impact advanced call features such as call forwarding and the ability to deploy configuration changes to CUCM / CUPS systems in a cluster. On the other hand, standard call operations are not affected. The CUCM vulnerabilities are documented in Cisco Bug ID CSCsg20143. The CUPS vulnerabilities are documented in Cisco Bug ID CSCsg60949 [19].

6.2.6.4. HTTP server DoS:

A denial-of-service vulnerability affects phones running the SCCP protocol. This vulnerability happens when handling the internal HTTP server. By sending a specially crafted HTTP request via TCP Port 80 to the affected devices the attacker could crash phones running SCCP, and this will cause affected devices to reboot. The motivation of an attacker behind this is to execute arbitrary code with super user privileges or crash the affected device, denying service to legitimate users. The vulnerability is being tracked by CVE-2008-0527 and Cisco Bug IDCSCsk20026 [18].

7. FURTHER WORK

For future work, more analysis is required for security problems on both systems and simulations need to be done for proposed solutions, measuring the solution effectiveness and performance stability of both systems.

8. CONCLUSION

In this paper we addressed the security issues arising in the main VoIP systems available on the market. We compared SIP and Cisco SKINNY to ensure confidentiality, integrity and authentication requirements. We have presented several attacks that will stop or cause high damage to services and performance in both studied systems. Those vulnerabilities which both systems suffer from are our comparison and the judgement between them. We conclude that both systems suffer from different security problems and the selection between them may need to involve other points, such as cost, scalability and support. Our analysis uncovered several serious vulnerabilities. The first is a replay attack on SDES key exchange which causes SRTP to use the same key stream in multiple sessions, thus allowing the attacker to remove encryption from SRTP-protected data streams. The second is an attack on ZRTP caused by unauthenticated user IDs, which allows the attacker to disable authentication mechanisms and either trick a ZRTP participant into establishing a shared key with the attacker, or cause the protocol to terminate prematurely. The third is a "certification" issue.

9. ACKNOWLEDGMENTS

The authors would like to thank King Fahd University of Petroleum and Minerals (KFUPM), Saudi Arabia, under grant no. RG1106-1, and the Hadhramout Establishment for Human Development, Yemen, for their support and providing computing facilities during this work. The authors would also like to thank the anonymous reviewers for their constructive comments which helped to improve the quality of the paper.

REFERENCES

[1] E. Coulibaly and L. H. Liu, "Security of Voip networks," in Computer Engineering and Technology(ICCET), 2010 2nd International Conference on, vol. 3, pp. V3–104 –V3–108.http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5485790&isnumber=5485220

[2] Z. Anwar, W. Yurcik, R. E. Johnson, M. Hafiz and R. H. Campbell, "Multiple design patterns for voice over IP (VoIP) security," in Performance, Computing, and Communications Conference, 2006. IPCCC2006.25th IEEE International, 2006, p.8pp.–492. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1629443&isnumber=34183

[3] Y. Lu and Y. Zhu, "Correlation-Based Traffic Analysis on Encrypted VoIP Traffic," in Networks Security Wireless Communications and Trusted Computing (NSWCTC), 2010 Second International Conference on, 2010, vol. 2, pp. 45-48. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5480425&isnumber=5480332

[4] E. Kokkonen and M. Matuszewski, "Peer-to-Peer Security for Mobile Real-Time Communications with ZRTP," in Consumer Communications and Networking Conference, 2008. CCNC 2008. 5th IEEE, 2008,p.1252. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4446592&isnumber=4446298

[5] C. Y. Yeun and S. M. Al-Marzouqi, "Practical Implementations for Securing VoIP Enabled Mobile Devices," in Network and System Security, 2009. NSS '09. Third International Conference on, 2009, pp.409–414. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5319311&isnumber=5318887

[6] M. Baugher, D. McGrew, M. Naslund, E. Carrara and K. Norrman. "The Secure Real-time Transport Protocol (SRTP)". IETF RFC 3711, March 2004.

[7] M. Baugher, D. McGrew, M. Naslund and K. Norrman. Secure RTP Frequently Asked Questions. http://srtp.sourceforge.net/faq.html, October 2005.

[8] P. Zimmermann, A. Johnston and J. Callas, "ZRTP media path key agreement for secure RTP, IETF internet-draft, work in progress, draft-zimmermann-avt-zrtp-15," Mar. 2009. [Online].

[9] C. Y. Yeun and S. M. Al-Marzouqi, "Practical Implementations for Securing VoIP Enabled Mobile Devices," in Network and System Security, 2009. NSS '09. Third International Conference on, 2009, pp. 409-414.
[10] V. Gurbani, S. Lawrence and A. Jefferey, "Domain certificates in the session initiation protocol (SIP)," IETF internet-draft, work in progress, draft-ietf-sip-domain-certs-04," May 2009.
[11] S. Lawrence and V. Gurbani, "Using extended key usage (EKU) f session initiation protocol (SIP)", IETF RFC 5924 May 2009.
[12] E. Y. Chen and M. Itoh, "A white list approach to protect SIP servers from flooding attacks," in Communications Quality and Reliability (CQR), 2010 IEEE International Workshop Technical Committee on, 2010, pp. 1-6.
http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5485790&isnumber=5485220
[13] R. Bresciani and A. Butterfield, "A formal security proof for the ZRTP Protocol," in Internet Technology and Secured Transactions, 2009. ICITST 2009. International Conference for, 2009, pp. 1-6.
http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5402595&isnumber=5402499
International Journal of Network Security & Its Applications (IJNSA), Vol.4, No.4, July 2012 152
[14] The Register, "Fugitive VOIP hacker cuffed in Mexico," http://www.theregister.co.uk/2009/02/11/ fugitive VoIP hacker arrested/, February 2009.
[15] S. Duanfeng, L. Qin, H. Xinhui and Z. Wei, "Security mechanisms for SIP-based multimedia communication infrastructure," in Communications, Circuits and Systems, 2004. ICCCAS 2004. 2004 International Conference on, 2004, vol. 1, pp. 575-578 Vol.1.
http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1346199&isnumber=29624
[16]F. Cao and C. Jennings, "Providing response identity and authentication in IP telephony," in Availability, Reliability and Security, 2006. ARES 2006. The First International Conference on, 2006, p. 8 pp. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5480425&isnumber=5480332
[17] http://securitytracker.com/id/1017826
[18] http://www.cisco.com/warp/public/707/cisco-sa-20110777-cucm.shtml
[19] http://www.cisco.com/en/US/docs/voice_ip_comm/cucm/admin/7_1_2/ccmcfg/b03dpi.html#wp1075 124.
[20] http://www.cisco.com/warp/public/707/cisco-amb-20110427-cucm.shtml

AUTHORS

Dr Talal Al-Kharobi received a B.Sc. degree In Computer engineering from the Department of Computer Engineering King Fahd University of Petroleum & Minerals(KFUPM) on 1988/93, and the M.S. He received a PhD. in computer engineering from The Texas A&M, College Station, Texas, USA 1998/2004. He is currently a professor in the Dept. of Computer Engineering, King Fahd University of Petroleum & Minerals (KFUPM). He has published many conference papers in wireless and networks protocols .

Mohammed Abdullah AL-Mehdhar received a B.Sc. degree from the Department of Computer Science, Hadramout University of Science and Technology between 1999/2003. He is currently a Master student in the Department of Computer Engineering, King Fahd University of Petroleum & Minerals(KFUPM). His research interests are IPv6 and Network protocols, multihoming protocols and applications, and Smart Grid.