Global Information Assurance Certification Paper

Title: Session Initiation Protocol (SIP) Hottest new technology in communications

Author: Sherif Makram Saad

Version: GSEC Practical Assignment (v.1.4b) (Option1)

24 January, 2004

INDEX and Table of contents

Session Initiation Protocol (SIP)
Hottest new technology in communications

Abstract:

The objective of this paper is to describe services that are integrated with Session Initiation Protocol (SIP) and to discuss its security. It also aims to understand how the protocol interacts with firewalls,
NAT (Network Address Translation) & Integration between DNS (Domain Name System) and SIP.

SIP designed by IETF (Internet Engineering Task Force) (RFC 2543 updated 3261) resides at the application layer of the network for the new standard for person to person IP-based real time communication as an application layer signaling protocol for creating, modifying, and terminating user sessions. These user sessions can be between two or more users, users here can be any agent using the user agent program (UA). SIP is used for Internet multimedia, conferencing, instant messaging and it can be used for IP Telephony (voice and video) and more…

SIP looks like the following e-mail address example:

Sip:Alice@domain.com
Sip:+123456789@domain.com
Tel: +123456789

Setting up sessions between users present big problems when the users are on private address space (NAT) or when the users are protected by a firewall. So the only way to overcome this problem is to implement SIP transparent Firewalls and NAT, SIP proxy and SIP registrar which dynamically control the firewall in addition to SIP aware Firewall & NAT**.**

SIP Definition & Benefits

Session Initiation Protocol was developed in the mid-1990s by the Internet Engineering Task Force as a real-time communication protocol for IP voice, and has expanded into video and instant-messaging applications. Ref##

SIP is simple, scalable, built around existing Internet environment.
SIP (Session Initiation Protocol) is IETF protocol (RFC 3265) designed with the idea that it is not just a session between end points. It is also designed to locate the persons and sets up person to person sessions even when the user changes terminals. Unlike other protocols performing similar functions, SIP is a generalized protocol designed for scalability, and relies on Internet services already in place such as DNS (Domain Name System) & H.323 protocol.
SIP include instant message via text channels with a unique new service that gives us a globally reachable address i.e. email address.

SIP is the leading protocol for Voice over IP and is replacing H.323 in this role. The reasons behind SIP existence is that H.323 which became complex, limited in functionality and also has proved poor scalability & reliability.
Therefore H.323 is expected to be obsolete when SIP is taking over rapidly during the last years. Besides, SIP is growing in popularity due to its ability to easily combine voice and Internet-based services.

SIP is independent of the packet layer and only requires an unreliable datagram service, (UDP) as it provides its own reliability mechanism.

SIP Benefits

The benefits for developing and working with SIP are as follows:

Simplicity, Modularity, Scalability, Integration with existing infrastructure, Ease of Implementation & is Customizable.

General Purpose

SIP is a text-based protocol, similar to HTTP (Hyper Text Transfer Protocol) and SMTP (Simple Mail Transfer Protocol) for initiating interactive communication sessions between users. These sessions include: voice, video, chat & interactive applications such as games.

SIP Servers need to deal with varying network topologies (public Internet Networks , broadband, residential networks beside complex routing policies, security & Mobile GSM Networks for Instant Messages (IM)…)

SIP Servers often need to handle high message transaction rates, real-time performance, scalability, high throughput, and considering low delay transmission. SIP mainly establishes connection by IP addresses and port numbers at which the other party systems can send and receive data.

SIP is a textual client-server base protocol that provides the necessary protocol mechanisms for the end user systems and proxy agent server to be able to provide different services.

SIP addresses Uniform Resource Locator (URL) can be embedded in Web pages and therefore can be integrated as part of powerful implementations (Click to talk, for example) [16]

SIP Components & Terminology

- Real Time Streaming Protocol (RTSP) which is an application-level protocol for controlling delivery of data with real-time properties.

- Real Time Protocol (RTP) must traverse a NAT to enable communication between endpoints that runs over UDP (User Data Program) and has no fixed ports associated with it.

- Real time Transport Control Protocol (RTCP) that will exchange information about [Sender, Receiver, Source description i.e. (name, e-mail, ID), Application defined] and also exchange information about delay between end systems & separate packets sent on different ports.

- Session Description Protocol (SDP) describes multimedia sessions for the purpose of session announcement, session invitation and other forms of multimedia session initiation. It also provides information about a call, such as the media encoding, protocol port number, multicast addresses , contact information & session name and purpose …

SDP is entirely textual data format rather than protocol. This means that the information carried by SDP is not coded into a more compact form i.e. ASN.1. (Abstract Syntax Notation 1) that is used by H.323 protocol for encoding.

SIP message body contains the information that the endpoints need in order to communicate directly with each other. This information is contained in the SDP message. I.e. User Agent (UA) located behind NAT knows only its internal IP address. and that is why it is put in the SDP body of the outgoing SIP message. Then the destination endpoint will start sending packets to the originating endpoint. It will use the received SDP information containing the internal IP address of the originating endpoint. Thus the SDP will hide the originator IP address and only the proxy will know real IP address. So, for security design of the SDP the UA will never need to know the public IP address of the collie.

SDP text messages include:

- Session name and purpose
- Duration of the active session
- Media comprising the session
- Information to begin receiving the media

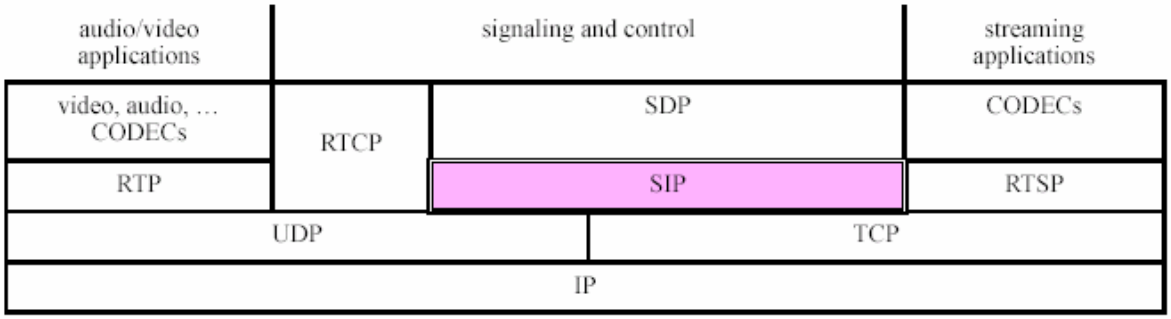


Figure shows SIP, RTP, RTCP & RTSP [5]

SIP requests Methods:

- INVITE
  The INVITE method means that the UAC (User Agent Client) is being invited to participate in a requesting session. The message body contains a description of the session to which the UAC is being invited.
- ACK
  The ACK method requests confirmation that the UAC on the other side has received a final response to an INVITE request.
- OPTIONS
  The UAS (User Agent Server) that delivers the request to the other UAC, will discover features supported by the receiver and requests information on the capabilities that are provided by the server.
- BYE
  The UAC agent uses BYE to indicate to the UAS that is requesting the call to be ended.
- CANCEL
  The CANCEL request cancels a pending request with the same Call-ID.
- REGISTER
  A client uses the REGISTER method to register the address listed in the (To) header field with the SIP server.
- Notify / INFO
  Sent by registrar server to notify the UAC or UAS about the URI (Uniform Resource Identifier) state whether changed or not.

Address Header Fields

1. *From*: message originator
2. *To:* final recipient
3. *Request-URI*: destination
4. *Contact:* appears in INVITE / OPTIONS / ACK / REGISTER requests.

SIP Message Structure:

Request Method	**Response Status**
INVITE sip:Bob@domainb.com SIP/2.0 Via: SIP/2.0/UDP domaina.com:5060 From: Alice<sip:alice@domaina.com> To: Bob <sip:bob@domainb.com> Call-ID: 123456@domaina.com CSeq: 1 INVITE Subject: Test Contact: Alice<sip:alice@domaina.com> Content-Type: application/sdp Content-Length: … v=0 o=Alice IN IP4 1.2.3.4 s=Test. c=IN IP4 alice@domaina.com	SIP/2.0 200 OK Via: SIP/2.0/UDP domaina.com:5060 From: Alice <sip:alice@domaina.com> To: Bob <sip:bob@domainb.com> Call-ID: 123456@domaina.com CSeq: 1 INVITE Subject: Test Contact: Bob <sip:bob@domainb.com> Content-Type: application/sdp Content-Length: … v=0 o=Bob IN IP4 5.6.7.8 s=Test c=IN IP4 bob@domainb.com

Summary of SIP response codes: [2]

- 1xx Status: **Informational** for request received and in the processing
- 2xx Status: **Success** the action was successfully received , and accepted
- 3xx Status: **Redirection** will be forwarded to another UAS to handle the request
- 4xx Status: **Client Error** the request contains bad syntax
- 5xx Status: **Server Error** the server failed to validate the request
- 6xx Status: **Global Failure** the request cannot be accepted by any server

SIP Detailed Response Codes: [2]

Informational	100 Continue 180 Ringing 181 Call Is Being Forwarded 182 Queued 183 Session Progress
Success	200 OK
Redirection	300 Multiple choices 301 Moved permanently 302 Moved temporarily 305 Use Proxy 380 Alternative Service
Client Error	400 Bad request 401 Unauthorized 402 Payment Required 403 Forbidden 404 Not Found 405 Method Not Allowed 406 Not Acceptable 407 Proxy Authentication Required 408 Request time-out 409 Conflict 410 Gone 411 Length Required 413 Request Entity Too Large 414 Request-URI Too Long 415 Unsupported Media Type 420 Bad Extension 480 Temporarily Unavailable 481 Call-leg/Transaction does not exist 482 Loop detected 483 Too Many Hops 484 Address Incomplete 485 Ambiguous 486 Busy Here
Server Error	500 Server Internal Error 501 Not Implemented 502 Bad Gateway 503 Service Unavailable 504 Gateway Timeout 505 SIP Version Not Supported
Global Failure	600 Busy Everywhere 603 Decline 604 Does not exist 606 Not acceptable

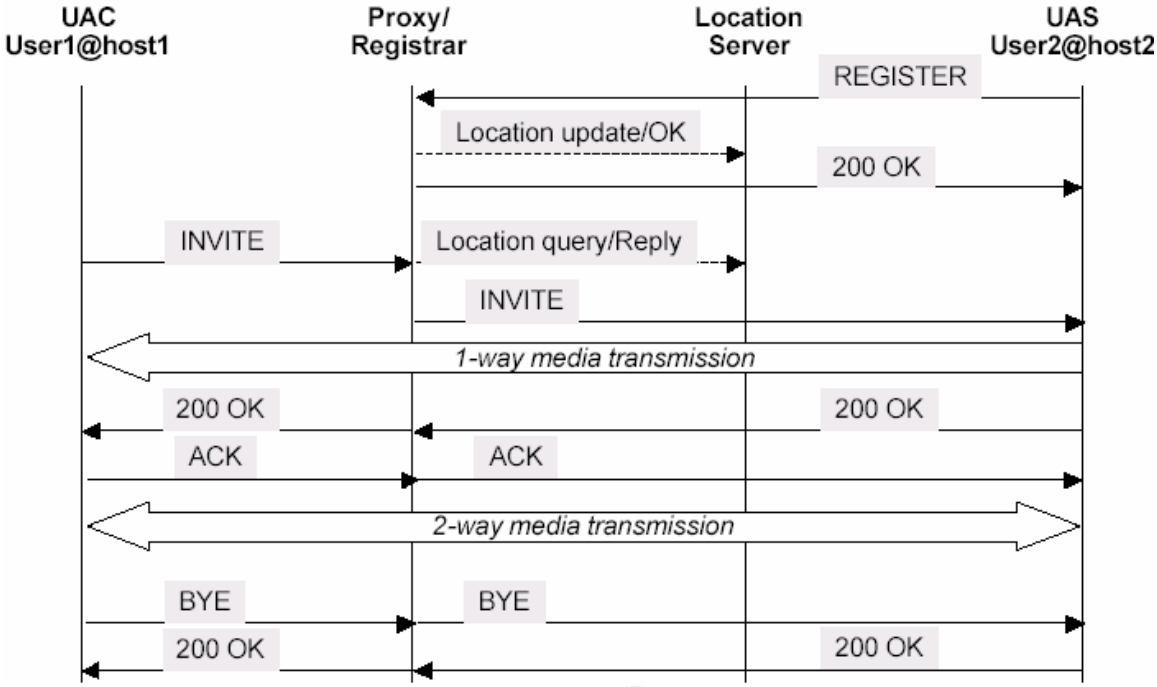


Figure shows SIP Message Structure [14]

The SIP baseline specification RFC3261, previously RFC2543 divides *SIP Server functionality* into the following parts:

- SIP Redirect /Location Server: redirects or locates callers to other servers and relays information to caller about the other caller location. It directs the client to contact another SIP address. The result in response returned as contact headers in the response message.

- SIP Proxy Server: relays call signaling, i.e. acts as both client and server and relays all messages between the caller and the other caller. A request may traverse several proxies on its way to a User Agent Server which will be described below.

SIP Proxies can be used in different purposes and in different locations in the Network such as: edge proxy, core proxy and enterprise proxy. Also a proxy server is designed to be almost transparent to user agents with restrictions to not change messages i.e.:

1. Proxy is **not allowed** to modify the SDP body of an INVITE request.
2. Proxy **cannot** generate requests by simply sending INVITE request.
3. Proxy **cannot** terminate an existing call by generating a BYE request.

SIP specification defines two types of SIP proxies:

1. **Stateful proxy**:

A stateful proxy processes transactions rather than individual messages. the proxy manages two types of transactions

- Server transactions to receive requests and return responses
- Client transactions to send requests and receive responses.

A stateful proxy is aware of the state of transactions and message history, and can therefore keep track of messages and retransmit incoming messages.

Disadvantages:

- **Memory consumption :**

- As it keeps record of transmitted messages which limits the number of concurrent calls/transactions it can handle.
- Negative impact on the maximum capacity of the proxy.

- **Throughput**

More CPU power consumption for message processing, managing transaction, and processing transaction which end to poor performance.

2. **Stateless proxy:**

A stateless proxy is just a message forwarder which means when it receives a request it forwards the message without saving any transaction context. This means that once the message is forwarded the proxy forgets about this message and waits for the following one. So, there is no way to retransmit any previously sent messages.

- SIP Registrar:

Accepts registration requests from users and places the information it receives into the location service for the domain it is responsible for. Register requests are generated by clients in order to establish a mapping between external known SIP address and the address they wish to be contacted at.
Registrar (resembles a DNS) matches the SIP address with the IP address by looking at DNS records for SRV with higher priority values. As we will get to this shortly in (Integration with SIP & DNS)

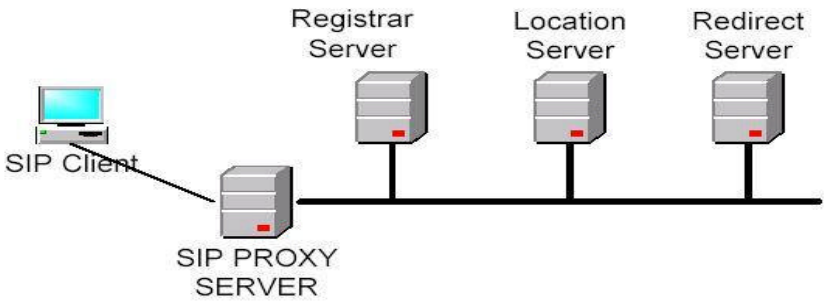
- User Agents:

The User Agent (UA) is an application program that runs at the user's machine.

1. User Agent Client (UAC)

2. User Agent Server (UAS)

UAC sends SIP requests on behalf of the user program application and the UAS listens for responses and notifies the UAC



[13]

Understanding the problems with SIP & Firewalls

Firewalls are essential to prevent & protect LAN (Local Area Network) from the Internet. Firewall is a barrier device placed between two separate Networks.

Most Common protocols such as HTTP, FTP, SMTP, Telnet have known port Numbers and can be managed by Network Administrator and this is done by putting rules in the firewall state where traffic is allowed (LAN, WAN, DMZ) and on which direction i.e. inside & outside connection.

The problem is that in SIP call, ports are allocated dynamically at every initiated call, The port numbers chosen are above reserved port range, in the range of 1024 - 65535 and may be different for every call. Therefore a firewall administrator can not know what ports to open for every call that will enable SIP to work. Therefore, the firewall administrator must open up nearly all the ports for successful communication. The Network administrator that is aware of security issues will not allow this to happen by simply opening bulk of ports. Most administrators do not like to do this and will only allow common services such as SMTP, FTP, Web Server & DNS if not security paranoid and will change default ports for those common application services.

Besides, normal firewalls will not let the SIP traffic through, since they do not know which ports to open for the voice traffic and at what time. For security reasons a large range of ports cannot be left open at all times.

SIP works on a well-known port 5060 but only for setting up sessions usually transmitted over TCP or UDP. So, SIP does not have problem direct with the firewall but it is the information that SIP has to have in the media stream for setting up the session between the participants, being the major problem. The ports that are used for the media streams are dynamic above 1023 reserved ports thus the firewall will not know that an audio stream destined for a certain address and port should be let inside/outside or not.

If we have closer look at RTP connection, that is a must for person to person real time communication, we will find that from inside firewall to the outside is possible but not vice versa.
UDP packets from outside will be blocked by the firewall, since they are not associated with an outgoing request as they are over a TCP connection.

Types of Firewalls

1. **Packet filtering** :

Low end Firewall filter for every inbound & outbound connection, configured rules are applied on per interface per direction. This is not recommended if we are considering defense, but it may be implemented as first level defense but not only the firewall that we will depend on for defending our networks. Always to consider defense in depth in security design.

2. **Application Level Gateway (ALG)** :

ALG provides best security and is referred to as proxy application gateway. The idea behind this type of the ALG firewall is that it will replace internal addresses with the address of the firewalls external interface.

We will discuss ALG for SIP in Design of a Secure SIP & Security Mechanisms section later.

3. **Satfull Packet Inspection** :

It is neither ALG nor Packet Filter. The Idea here is to examine each content of a packet and make sure it's what claims to be, recommended for most secure environments. Stateful inspecting firewalls also track commonly used application protocols that use multiple connections.

Understanding the problems with SIP & NAT

Network Address Translation (NAT) [RFC1631] translates an IP address used within one network to a different IP address known within another network. NAT works by changing a packets header of private IP address information to an public accessible address.

Reason behind NAT existence is:

1. The shortage of Real public IP-addresses lead to the need for NATs that allow the use of private IP-addresses behind a single or multiple public IP address.

2. Second reason the private addresses solve the big problem which is the rapid usage of the public IP addresses available.

Below is the range of the private addresses:

10.0.0.0 - 10.255.255.255 Class A networks
172.16.0.0 - 172.31.255.255 Class B networks
192.168.0.0 - 192.168.255.255 Class C networks

NAT creates problems for the SIP protocol Such as:

- Networks behind a NAT with private IP addresses cannot be accessed from the Internet. Thus, an incoming SIP call cannot directly be routed back to SIP device behind the NAT.

- NAT prevents two way multi media communication because the private IP addresses are not routable on the public Internet Network. Also different ports for private IP & Public IP means that if I'm behind a NAT and I'm using private virtual IP address, I'll connect to the NAT device first for delivering my packet with my own IP address and port No. Then the NAT breaks the connection and changes the source address with its own external IP. The NAT will then deliver my packet to destination server on the Internet. The public server when responding will contact the NAT external interface and change the destination IP to my IP private address. That's why there is no direct connection for media communication stream.

- NAT also prevents communication using RTP protocol i.e. when UA requires RTP with another UA there will be misrouted IP address in SDP header.

We will discuss the solution to solve this problem in Design of a Secure SIP & Security Mechanisms section later.

Types of Network Address Translation

- Static NAT: Permanent mapping between local addresses and global addresses.

- Dynamic NAT: Reduce global addresses used as dynamically assigned pool of private IP to External public IP.

- NAPT Network Address and Port Translation: (NAT & PAT) most used types in firewalls today that translate both IP address & Ports numbers from one network to another network. In such way this type of firewalls is limited per port No. per IP.

Basic Security concerns with NAT & Firewall

- Multimedia protocols don't necessarily know the source port of the media stream. This means firewalls would not be able to dynamically open ports for successful transmission.

- Most of firewalls & NAT devices currently in the market (Time of writing this paper) do not support Multimedia over IP Protocols.

- Firewall & NAT by nature do not allow inbound connection, unless otherwise configured by security administrators to do so for special purpose.

- Real-time Communications protocol (RTC) only can work with NAT that support Universal Plug and Play (UPnP) functionality , which enables audio & video to be transmitted via IP address and port that are located on the NAT device public accessible interface. If UPnP NAT is not available then the RTC will need to open two ports to Map IP addresses and ports on the NAT, one for initiating SIP call as previously mentioned and one for streaming media.

We will discuss the UPnP NAT in Design of a Secure SIP & Security mechanisms section later.

SIP Security Requirements & Vulnerabilities

"Numerous vulnerabilities have been reported in multiple vendors' implementations of the Session Initiation Protocol. These vulnerabilities may allow an attacker to gain unauthorized privileged access, cause denial-of-service attacks, or cause unstable system behavior." [15]

"The Computer Emergency Response Team Coordination Center (CERT/CC) has released an advisory warning of multiple vulnerabilities in Session Initiation Protocol (SIP) implementations from a variety of vendors.
The vulnerabilities could be exploited to launch denial-of-service attacks, gain unauthorized access to systems or cause system instability. Vendors are offering patched for the problems." [15]

1. Authentication:

Process of confirming claimed identity with several methods involves User ID & password, Digital certificate authorities, shared key, IR, Finger Prints…

2. Confidentiality:

Means only the intended recipient of a message will be able to know the contents. Confidentiality relies on several algorithm such as Triple DES, AES, MD5 …

3. Integrity:

Message integrity insures that a message was not altered in transit.

4. Authorization:

Once authentication of an identity been successful, then a decision made upon whether to grant or deny access.

5. Privacy:

Securing the Identity information using encryption to prevent unauthorized persons from inspection both signaling and media.

6. Non Repudiation :

By using Digital Certificates & Digital Signatures, certificate guaranteed that the user can't deny the message originated from him with integration with signature will verify that the SIP request is not altered during transmission. SIP requires the use of a public-private key encryption method such as RSA, Diffie-Hellman or PGP…

Threats Methods

1. Registration Hijacking

When the User Agent (UA) registers with the Registrar server , the Registrar server will check the message (From) field to verify the identity to deliver it to the (To) field. The (From) header could be modified by the owner of the message (Requester) This can open a back door that allows the attacker to change contact information or redirecting sessions to his (UA) device.

To prevent hijacking to exist we have to deploy strong Cryptographic mechanisms to ensure the message register originator.

2. Server Hijacking

When the User Agent (UA) contacts the server to deliver a request for a message there is a possibility that the server was compromised. That could lead to redirecting the request to another domain “which receives the SIP header” and simply ignores the request. This enables the receiver server to frog the connection and could indicate a false alarm.

To prevent hijacking to exist we have to deploy strong Authentication that enables the UA to authenticate with the server before sending any request such as RADIUS Authentication Server.

3. Message Hijacking

When the User Agent (UA) routes the message through proxy agent and this message is in encryption format, then the proxy can modify the session encryption key, MIME bodies that prevent the receiver server from decrypting it. Besides the proxy agent itself could be compromised that will break the security of the message that the UA was originally originating. That’s why as we mentioned before, the proxy should not modify any message by itself.

To prevent hijacking to exist we have to deploy strong (CIA) Confidentiality, Integrity, and Authentication at the proxy server level.

4. Hijacking Established Session

Once the connection is established, the attacker could sniff the initial parameters such as (To, From …) fields and then send a BYE request. In the same way, the attacker can do while compromising a system to not complete the three way hand shake (SYN / ACK / FYN) by only sending SYN then followed by FYN. This ends with unexpectable closing transmission channel. Then the attacker will send Re- INVITE request after knowing where is the target server. In this scenario the server does not have any information that the UA is got compromised. So, the server will continue the session assuming the connection is still alive.

To prevent hijacking to exist we have to deploy strong Authentication that will make sure that BYE is coming from one of both parties.

5. Denial Of Service Attack (DoS & DDoS)

Denial Of Service (DoS) focuses on directing network traffic. Distributed denial of service attack (DDos) enables one network host to flood target host with large amount of undesired traffic which leads to DoS. An attacker could spoof the source IP by modifying header field of the target host as if it's the requester.

Another type of attacking is that the attacker can redirect the routing header in the requested message that will mislead the proxy agent for routing requested message.

The Attacker also attempts to consume the registrar memory & Disk available resources to the server by registering huge numbers of messages requests that will prevent the registrar to respond to all the requests which leads to server flooding and buffer consumption.

The idea behind DDoS is sending requests to server(s) from distributed locations around the world at one time and by default the server(s) will try to respond to these requests which got the server out of production for being busy that leads to server crash.

To Prevent Dos to exist we have to deploy strong Confidentiality, Authentication, Authorization and Accounting (AAA). This is done at the Registers level.

Integration between SIP and DNS

DNS (Domain name system) is a well known service that Internet domains relies on for resolving & Map Domain names to IP address. SIP resembles mail server and has the same format like e-mail address that use DNS MX (Mail Exchange) records to identify mail server point back to IP address. In SIP case it relies on DNS lookups to identify proxies, ports, transport protocol of the next hop that will deliver a message from i.e. Domain A to Domain B.

SIP can integrate with web page & mail technologies that use DNS Uniform Resource Identifier (URI), HTTP requests for a route or destination IP address of top-level 13 domains and sub-domains to create a distributed and decentralized database.

So how can we use SIP to make phone calls? By using DNS query we can establish a connection that will look like e-mail address within domain name. The technique behind this is E.164 that handles numbers in DNS (ENUM) zone (which we will discuss shortly) which allows SIP servers & clients to send and receive telephone numbers in SIP URI format. (SIP URIs includes Telephone URLs)

It is not that simple if we compare it to normal telephone numbers, In DNS the SIP will use Reverse DNS number i.e. No.enum.164 plus SRV domain records. At the end the UA will never need to know how this mechanism is working. That means the SIP proxy will do this translation automatically.

In closer look for DNS mechanism here as we remember SIP is a text based protocol that will use the default structure in HTTP protocol and Simple mail transfer protocol (SMTP). So, for this to be done here is the role for RTC protocol that will help connecting people regardless of where they exist.

By using SRV in DNS, once the call request is received by UA it will try to locate SIP server by asking DNS server for SRV, A, CNAME records for the SIP server. Described in RFC 3263 which state that, if for any reason the above steps fail, the gateway (NAT / Firewall) will try to resolve the target SIP user by resolving the domain itself in DNS records. If domain can not be resolved then a generated failure report will be delivered to UA.

In SIP the method of dealing with DNS is different than mail exchange record which means callers initiate calls by asking DNS for high-priority server. If unavailable, they proceed with lower priority server.

SIP also integrates with DNS by using LDAP protocol (Lightweight Directory Access Protocol). We imagine that LDAP will become the phone directory services that will provide SIP with the information required for specific user through directory services or corporate Database.

If we are considering successful design for SIP & DNS we should consider DNS Round Robin for load balancing and fail-over. DNS servers maintain multiple SRV entries with equal values.

SIP URL example:

sip: User name@domain.com

sip: Phone Number@domain.gateway.com , a call from the Internet to the PSTN E.164

sip Phone Number@proxy.domain.gateway.com , proxy server determines gateway and forwards the request.

sip: User name@registrar.domain.com , UA =register a user at a SIP registrar.

SIP SRV example:

SIP clients use DNS SRV records if available. A sample DNS zone file entry is shown below:

sip. tcp	SRV 5060/tcp sip-server.domain.com
	SRV 5060/tcp sipbackup.ip-domain.com
sip. tcp	5061/tcp sip-server.domain.com {over TLS}
sip. udp	SRV 5060/udp sip-server.domain.com
	SRV 5060/udp sipbackup.ip-domain.com

ENUM & NAPTR

Designed by IETF E.164 (Number Mapping standard) that uses (DNS) to map standard International Telecommunication Union (ITU-T) to a list of Universal Resource Identifier (URI) SIP then uses those URL's to initiate sessions. For example:

ENUM DNS converts a telephone i.e. +123456 number in E.164 format to and returns (URI) i.e. SIP: username@domain.com
By using reverse DNS i.e. +123456 to 6.5.4.3.2.1.e164.arpa

Thus a SIP client makes a connection to the SIP gateway domain.com passing any value before @ symbol, in this case will be username. (Destination party) What is the value added that ENUM have, At closer look we will find that when we intend to call another party all we have to remember is a format that combines both email address with SIP address, So the Idea here is to simplify communication methods instead of remembering two different addresses, one for email and one for SIP calls. ENUM enables a user to store contact information, such as fax, voice, voice mail, email, Web and home address information.

The URI resource records used by ENUM are Naming Authority Pointers Records "NAPTR". DNS NAPTR records are used to allow a client to discover that the server supports TLS.

NAPTR fields contain a number of components**:**

* Preference field: Determine the processing order when multiple NAPTR records have the same order value.

* Service field: Specify the resolution protocol and service.

* Replacement field: Define the next DNS query object.

* Order field: Specify in which multiple NAPTR records must be processed.

Comparison between H.323 & SIP [9]

Comparison	H.323	SIP
Protocols	RAS/Q.931 H.245	SIP SDP
Origin	Extension of ITU-T standard H.320	IETF
Conference	Yes	Yes
Components	Terminal Gateway Gatekeeper Multipoint Control Unit	User Call Agent (UA) Proxy server Redirect server Registrar server
Philosophy	H.323 mainly designed for multimedia communication over IP networks, including audio, video, and data conferencing.	SIP mainly designed to setup a "session" between two elements of the Internet architecture. Using compatibility between web sites and browsers , emails … SIP hardly supports conferencing via INVITE; however, it can perform conference, RFC 3261 states that "SIP does not offer conference control services." [4]
Connection State	Stateful	Stateful or Stateless
Transport	TCP	UDP and can use any transport protocol
Reliability	H.323 has defined a number of features to handle failure of network devices.	SIP has not defined procedures for handling device failure.
Security	Use SSL for transport-layer security.	Supports security by the following methods : • Authentication via HTTP

		mechanisms. • Cryptographically secure authentication and • encryption is supported via SSL/TSL • SIP also defines end-to-end / Hop to Hop authentication/encryption by using either PGP or S/MIME.
Message Encoding	H.323 encodes messages in a compact binary format that is suitable for High & Low bandwidth connections.	SIP messages are encoded in ASCII text format, suitable for humans to read. As a consequence, the messages are large and less suitable for networks where bandwidth, delay, and/or processing are a concern.
Media Transport	RTP/RTCP	RTP/RTCP
Non Standard features	H.323 is extended with non-standard features in such a way as to avoid conflicts between vendors.	SIP is extended by adding new header that may be used by different vendors for different purposes, (Risk Exist)
Standard Extended	H.323 support standard features that, can be extended to new features in such a way as to not impact existing features.	SIP support standard features, can be extended to new features in such a way as to not impact existing features.
Load Balancing	H.323 has the ability to load balance endpoints across a number of gatekeepers devices. I n addition, endpoints report their availability and total capacity so that calls can distributed across multiple gateway.	SIP does not support load balancing. However it relies on DNS SRV. In the future will rely on HTTP protocol.
Scalability	Whenever H.323 gatekeeper is used, it may simply route all call signaling traffic to endpoints that connect directly to one another.	Whenever SIP proxy is used, the proxy is required to handle at least 3 full message exchanges for every call.
Address Resolution	H.323 defines an interface between the endpoint and gatekeeper i.e. ARQ or LRQ. The H.323 gatekeeper may use any number of protocols to discover the destination address, ENUM, DNS.	While SIP has no address-resolution protocol, SIP UA route INVITE message through a proxy or redirect server by using protocols such as ENUM, DNS.
Addressing	H.323 includes URLs and E.164	SIP only understands URL

mechanisms	numbers.	addresses.
Billing	H.323 in direct call model, there is an ability to successfully bill the call because the endpoint reports to the gatekeeper the beginning and end time of the call.	In SIP the only way to successfully bill the call, SIP in Proxy mode it has to stay in the call signaling path for the entire duration of the call.
Forking the Call	H.323 gatekeeper can control the call signaling and may fork the call to any number of devices simultaneously.	SIP proxies can control the call signaling and may fork the call to any number of devices simultaneously.
Video & Data Conferencing	H.323 fully supports video and data conferencing.	SIP has limited support for video and no support for data conferencing protocols.
Integration with Firewall	Provided by H.323 proxy	Provided by SIP proxy
Ports for VoIP Call	5 (Call signaling, 2 RTP, and 2 RTCP.)	5 (SIP, 2 RTP, and 2 RTCP.)

Design of Secure SIP & Security Mechanisms

First of all we need to implement Firewall with SIP-Aware NAT Router to trick a firewall into allowing a SIP session to be established.

Recommended Solutions:

- Application Layer Gateways (ALG)

ALG is an SIP and RTP proxy that is trusted by most of the firewalls

ALGs are installed on the firewall or on the NAT gateway. ALGs usually work at a lower level than a proxy, adjusting the data packets on spot.
ALG architecture cannot handle secure SIP signaling via TLS (Transport Layer Security).

The Firewalls should relay SIP traffic and keep track of which ports should be used for NAT, enabling machines on different sides of a firewall to send and receive media streams just as if there was no firewall at all.

By considering the following in design:

Considerations:

1) For outgoing SIP requests, only a SIP proxy is needed.

2) For Incoming request we need a device that will keep track of the local users to relay the incoming request to the assigned machine & user.

3) The SIP proxy server in the firewalls handles the SIP NAT combination by rewriting the SIP headers to give the right IP addresses. In this way Firewalls can relay SIP requests for a UA regardless where is the user located because in this scenario the SIP will use external accessible firewall interface.

4) Clients must be registered at the ALG. This will give the ALG the internal address of the client, whose name was found in the (To) field.

Drawbacks:

* Firewall ALG will fail to operate if data in encrypted format.
* NAT ALG will fail to operate if data encrypted or authenticated.

The Idea here is to split ALG from NATs / Firewalls or use SIP aware NATs / Firewalls.

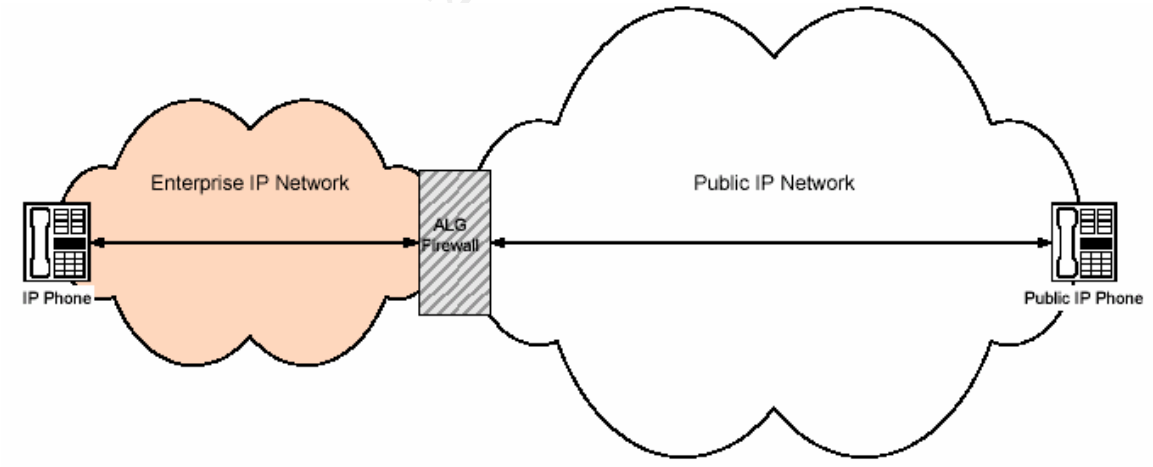


Figure shows Application Layer Gateways [13]

- Simple Traversal of UDP Through NAT (STUN)

STUN is a method for passing SIP signal through NATs by determining what type of NAT the client is behind and whether NAT exists or not .It works through keeping holes open at the NAT and changing the IP address of the SIP client in a way as if they are coming from out side the NAT with public IP by using a server located in the public Internet. STUN will not work for all NATs & firewalls, and may have some scalability and security issues. The SIP client has to implement STUN and integrate it in the SIP stack to make it work.

How STUN Works:

The STUN server will send its response to the IP-port specified in the (Response address) and If that field is not present, then the server sends its response to the IP-port that it received the (Requested from).

If both IP and Port flags are not set, the STUN server responds from the IP-port that the (Initial packet) was sent to.

If the change IP flag is set, the server replies from a different IP.
If the change Port flag is set, the server replies from a different port.

STUN Response Information: [3]

Mapped Address: IP-Port of the client at STUN server outside the NAT.

Changed Address: Source IP address of the returned response.

Source Address: IP-Port that will be used by the client in returning response.

- Tunneling Techniques

This method requires tunneling media signaling through Firewall & NAT with proxy residing behind the Firewall/NAT to a public server. We can achieve this by installing two new servers, i.e. one on the private network tunneling with the second public server. This tunnel will carry the SIP signal. Let's see how this is done:

The public server will accept the SIP traffic then will modify IP-port address space then passes it to the Firewall. The firewall is already configured to accept traffic from that server. Then the firewall will pass it to internal server and vice versa.

The VPN tunnel is not encrypted by defaults but if we are considering security issues we are should implement IPSec. That is because if the public server has got compromised, then there is a chance to affect the firewall that already accepts traffic from that server and passes it to internal network server.

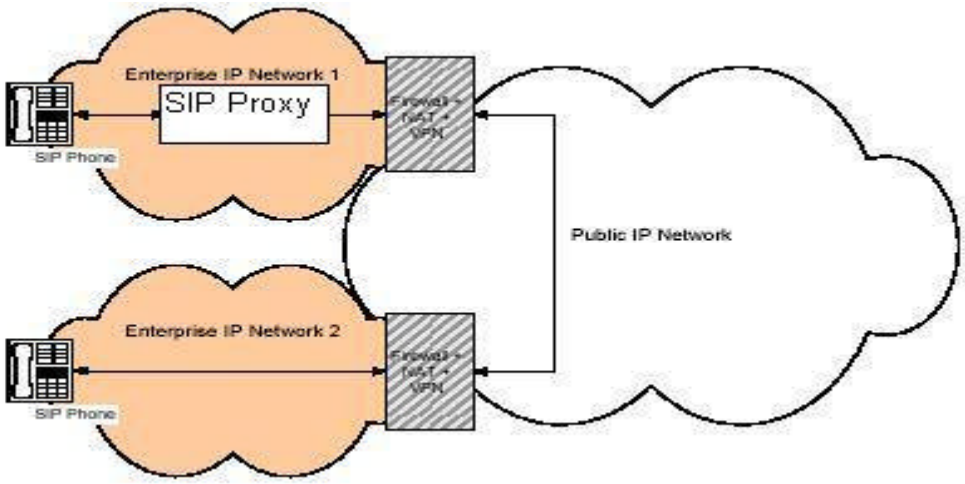


Figure shows Tunneling Techniques [14]

- UPnP (Universal Plug and Play)

This method solution is designed by Microsoft corporation for Small Office Home Office (SOHO) Networks which already exist in Microsoft Operating Systems and new MSN Messenger that support SIP protocol. The basic idea here is that the client will require the NAT device via UPnP to map public IP and port number on the external interface that is already accessible by another collie that requires SIP session, to the requester client.

UPnP allows clients application to discover and configure network components including Firewall & NAT by means of RTC Client API software that will use the information collected in paragraph one to establish VoIP calls.
This will ensure that the call is using public routable IP-port for successful end to end connectivity.

The Problem here is that UPnP relies on NAT and will not be efficient if the NAT device has no accessible IP address on the Internet side. Or if the ISP provider is giving already a bulk of NATed IPs to the customers, or even if we have cascaded NAT devices.

Even if we have no other solution it is not secure to allow every PC on the LAN to open port on the firewall that should protect the LAN, So UPnP is not acceptable when we are concerned about security issues.

- Static Route of Firewall Configuration [17]

Setting up a static route on the NAT gateway is the most powerful, but also most complicated way of setting up the phone in a NAT environment.

This method requires the client to must have static IP & Port for receiving media signaling.

1. Dynamic RTP port start, end: The range of ports that are used by the phone for media including start and excluding the end port.

2. Network identity (hostname, port): The SIP will insert name as hostname and port into the SIP messages. These values must match the router setup.

3. Local SIP port: With this flag we can decide whether the SIP uses the standard port (5060) or the port provided in the network identity as local port. If the router is not able to translate ports, we must use network port translator (NAPT)

4. RTP Media Traffic: We have to keep in mind that in this scenario we should use the other party router, to set up one UDP port for the SIP traffic and several ports for the RTP (media) traffic. As we already know that media streaming is located on dynamic port.

- NAT Proxy & RTP Relay

Sometimes for most of the NAT devices will not allow SIP calls, So, here is why we should consider design RTP relay in the middle of the RTP connection oriented between endpoints. At this stage we should consider also NAT proxy combining with RTP relay. NAT Proxy role here is to instruct the endpoints to send RTP to the RTP Relay instead of sending RTP direct to each other.

The RTP relay once received the request will set its own mapping on behalf of the endpoints and send it to the RTP.
We can also enable modification in the SDP header that would make the recipient RTP ignoring the IP address in the SDP which is not routable.

Drawbacks:

1. The client will always need to send and receive RTP on the same port.
2. There should be delay in this solution, so we should consider increasing the bandwidth.
3. The client will not hear any voice until the first packet is sent to the RTP Relay that will enable it to determine the public IP to start the session.

- General Considerations for SIP Secure Design

For successful Firewall & NAT implementation:

1. Always initiate outbound connections for inbound calls.
2. Always create NAT bindings, inbound traffic follows the return path
3. Always use a very small set of well known ports instead of dynamic random ports.
4. Always implement reachable SIP address for UAs by ensuring the validity of the DNS domain.

SIP Encryption & Security

SIP Authentication:

- **Basic & Digest :**

Basic and Digest is based on shared secrets between the client and the server. Basic is not secured since the passwords are transmitted over the network in clear text.
Digest is based on cryptographic hash of a number of elements. These elements include a username, password and a challenge provided by the server.

- **Security with PGP:**

Pretty Good Privacy is based on public key cryptography in a way the message encrypted and then digitally signed. The idea here is that the user composes a message with hash similar to the Digest mechanism, and uses his private key for encrypting & digitally sign the message. The signature calculation is then included in the SIP message header.

- **Security Using S/MIME:**

S/MIME (Secure/Multipurpose Internet Mail Extensions) provides a cryptographic security services to send and receive secure MIME data and treats SIP message like email attachment.

- **Security using RADIUS:**

By using RADIUS the UA authenticate itself with the proxy by using HTTP digest authentication. The proxy in this scenario will perform RADIUS client and send the user name & password to RADIUS server. RADIUS Role is to retrieve the password from database and computes the hash from the password.

If the computed hash matches the hash received from the proxy then the client granted access permission.

- **Digital Signature:**

The sender sends a message with hash code encrypted by CA (Certificate Authority) that issues certificate with private key and the recipient decrypts the hash code message using sender's public key.

However all above methods of authentication do not guarantee that the SIP message has not been modified or altered during transmission.

Encryption Considerations: [2]

Hop to Hop encryption

Encrypts entire SIP message and is supposed to work on the transport level or the network layer. Both IPSec (IP Security Protocol RFC 2401), TLS (Transmission transport layer protocol RFC 2246) are recommended in this case.

To achieve most secure environment in Hop to Hop connection:
We have to follow these guide lines:

1. we **must** allow source filtering at Firewall level.
2. we **must** block unauthorised sources and protocols to pass the network.
3. we **must** allow media ports to be dynamically opened from inside network to outside destination.

End-to-End encryption

When using end-to-end encryption between user agents we have to consider the following:

1. All header fields in message body **must not be** encrypted to be understood by intermediate Redirect SIP server.
2. An encryption header **must be** inserted to indicate the encryption mechanism.
3. The headers that were encrypted in the request **should** also be encrypted in the response.
4. SIP traffic that is sent over a public IP network **must be** encrypted.

SIP Disadvantages

1. SIP does not support media transport protocol.
2. SIP does not depend on a certain compression.
3. SIP does not support conference control protocol.
4. SIP proxies can not usually control media path as there is split between signaling and media. (Port to initiate request and port for media stream)
5. SIP does not provide QoS support. So there is no quality of services grantee.
6. SIP does not support proxy-to-proxy authentication.
7. NATs and firewalls generate serious problems for SIP sessions.
8. SIP data can be sniffed when corporate network contains Hubs.
9. The SIP standard does not define a specific type of encryption or authentication technique that must be used by SIP implementations. It does however specify how HTTP authentication and PGP encryption can be used with SIP, and leaves it open for other authentication and encryption mechanisms as well.
10. We can't rely on DNS as most of the corporate do not host domain on there servers or they do not have domain registered yet, they only have Internet connectivity.

Summary & Conclusion

The Session Initiation Protocol (SIP) is one of the hottest new technologies in communications, enabling a revolution in applications and deployment models.

The Session Initiation Protocol (SIP) is a signaling protocol used for establishing sessions in an IP network. Session could be a two-way telephone call or it could be a multi-media conference session. Over the last couple of years, the Voice over IP has adopted SIP as its protocol of choice for signaling, and the industry has focused a great deal of attention on this emerging standard. Currently, SIP is a draft from the Internet Engineering Task Force (IETF).

SIP is a request-response protocol that closely resembles two other Internet protocols, HTTP and SMTP (the protocols that power the Web and email). Using SIP, telephony becomes another Web application and integrates easily into other Internet services. SIP is a simple toolkit that service providers can use to build voice and multimedia services.
SIP is still being extended as technology matures through newer RFCs: #3261 ~ #3265, still SIP products are being developed in the marketplace today.

References

[1] Dynamicsoft - Rosenberg, Jonathan Chief Scientist
"SIP and NAT" & "SIP Security" 07/11/02
URL: http://www.dynamicsoft.com/news/presentations/SIPnNAT.pdf
URL: http://www.dynamicsoft.com/news/presentations/SIPSecurity.pdf

[2] Columbia University, Thernelius Fredrik "Master's Thesis
SIP, NAT, and Firewalls" May 2000
URL: http://www.cs.columbia.edu/sip/drafts/Ther0005_SIP.pdf
URL: http://www.networksorcery.com/enp/protocol/sip.htm#Description

[3] DeltaThree. Sterman, Baruch and Schwartz, David "NAT traversal in SIP"
2001 URL : http://corp.deltathree.com/technology/nattraversalinsip.pdf

[4] SIP RFC 3261 "Session Initiation Protocol" & RFC 3263 "Locating SIP
Servers" URL: http://www.ietf.org/rfc/rfc3261.txt
URL: http://asg.web.cmu.edu/rfc/rfc3263.html#sec-2
URL: http://www.faqs.org/rfcs/rfc2543.html

[5] Maguire, J.Q "2G5564 Practical Voice Over IP (VoIP) SIP and related
protocols" January 23 2003
URL: http://vvv.it.kth.se/edu/Ph.D/2G5564/VoIP-20030226.pdf

[6] SIP Server Toolkit version 1.0, Publication 1 "Understanding SIP Servers"
URL: www.sipcenter.com/files/RADVISION_Understanding_SIP_Servers.pdf
August, 2002

[7] Intertex Data AB, Lars Berggren and Karl Erik Stahl, 2 December 2001
Ingate Systems AB, Lisa Hallingström and Janne Magnusson,
11 October 2001 "The SIP Protocol and Firewall Traversal"
URL: http://www.intertexdata.com/upfiles/IntertexSIPWhitePaper.pdf
URL: http://www.ingate.com/files/sipwp-en-30.pdf

[8] GMD-Fokus Mobile Integrated Services, Kuthan Jiri and Sisalem Dorgham,
"Understanding SIP" 24 April 2001, URL: http://iptel.org/sip/siptutorial.pdf

[9] Microtronix system TLD & Packetizer, Inc. "SIP Vs. H.323 - A Comparison"
URL: http://www.microtronix.ca/sip_vs_h323.htm
URL: http://www.sipcenter.com/aboutsip/siph323sipandh323.html
URL: http://www.packetizer.com/iptel/h323_vs_sip/

[10] Columbia University, Schulzrinne Henning "SIP Security" 15/01/2002
URL: http://www.ietf.org/proceedings/01dec/slides/sip-7/sld001.htm

[11] Ridgeway Systems & Software Co, Davies Steve CTO “Boundary Traversal with IPFreedom” April 2003
URL: http://vcc.urz.tu-dresden.de/Projektkalender/ws_2003-04-10/IPFreedom%20in%20DFN.pdf

[12] Community Grids Laboratory, Indiana University, *Wenjun Wu, Ahmet Uyar, Hasan Bulut, GeoffreyFox*
“Integration of SIP VoIP and Messaging with the Access Grid and H.323 Systems” December 2003
URL: http://grids.ucs.indiana.edu/ptliupages/publications/sip-webservices-short02.pdf

[13] RADVISION Technology White Paper , 2001
“Traversal of IP Voice and Video Data through Firewalls and NATs”
URL: http://www.h323forum.org/papers/firewall_nat_traversal.pdf

[14] Soitinaho, Jouni, “Session Initiation Protocol” 5 April 2001
URL: http://keskus.tct.hut.fi/opetus/s38130/k01/Slides/Soitinaho-SIP-Slides.pdf

[15] Cert Advisory CA-2003, Original release date: February 21, 2003,
URL: http://www.cert.org/advisories/CA-2003-06.html

[16] VoIP Protocols URL http://www.protocols.com/pbook/VoIPFamily.htm#SDP

[17] ABP International Inc. , March 9 2003
URL: http://www.iptel.org/info/products/etc/snom-stun.pdf

[18] Martinez, Robert “SIP Security” 2003
URL: http://smuhandouts.com/8393/Security-Martinez.pdf