VoIP SECURITY

Rahul Singhai[*] **Prof. Anirudha Sahoo**[†]

Abstract

Voice over Internet Protocol (VoIP) technology has come of age and is quickly gaining momentum on Broadband networks. VoIP packetizes phone calls through the same routes used by network and Internet traffic and is consequently prone to the same cyber threats that plague data networks today. These include denial-of-service attacks, worms, viruses, and hacker exploitation.

The security concerns associated with IP telephony-based networks are overshadowed by the technological hype and the way IP telephony equipment manufacturers push the technology to the masses. History has shown that many other advances and trends in information technology (e.g. TCP/IP, Wireless 802.11, Web Services, etc.) typically out-pace the corresponding realistic security requirements that are often tackled only after these technologies have been widely adopted and deployed.

This paper explains the security risk factors associated with IP telephony-based networks and compares them, when appropriate, with the public switched telephony network (PSTN) and other traditional telephony-based solutions. It also outlines steps for helping to secure an organization's VoIP network.

Keywords: VoIP, Threats, Security.

1 Introduction

VoIP is one of the hottest trends in telecommunications. Before VoIP, telecommunications occurred over a public switched telephone network (PSTN), that is, voice data traversed circuit switched connections. The cost savings of Internet telephony systems by converging voice with other data applications, both in dollars and bandwidth, compared to that of circuit switched networks, is encouraging companies to move to VoIP. But many companies are unaware of the additional security baggage that voice brings along with it.

Once voice is converged with data on the network, a company's voice systems are suddenly vulnerable to many of the same kinds of attacks that occur on the data side. Phones can suddenly become destinations for spam. Hackers can target phone systems with denial of service attacks, or program a company's phones to call other businesses, shutting down the second company's phone systems. People can spoof a phone's IP address and make calls that are billed back to the company. And as with a traditional phone system, calls can be intercepted and listened to.

VoIP security is complicated by the requirement of multiple components, in most cases, more components than traditional circuit switched networks, and the fact that it is normally deployed on the current data network. Often, normal deployment requires co-existence of the circuit switched network until VoIP functions have replaced those of the circuit switched network. The security approach taken should address circuit switched network and VoIP for as long as both exist.

VoIP's next big step is toward wireless. Phones that can roam between Wi-Fi and cellular systems are on the way and will place further roaming and security challenges on VoIP systems.

2 Voice over IP

Voice over Internet Protocol is the routing of voice conversations over the Internet (*Voice on the net, VON*) or any other IP-based network (*Voice over IP, VoIP*) [Ark02]. The voice data flows over a general-purpose packet-switched network, instead of traditional dedicated, circuit-switched voice transmission lines. VoIP traffic might be deployed on any IP network, including ones lacking a connection to the rest of the Internet, for instance on a private building-wide LAN.

Protocols used to carry voice signals over the IP network are commonly referred to as *VoIP protocols.* There are currently three protocols widely used in VoIP implementations the H.323 family of protocols, the Session Initiation Protocol (SIP) and the Media Gateway Controller Protocol (MGCP). A basic difference between these three protocols is where intelligence is concentrated. SIP places most of the intelligence at the endpoints of the system. MGCP places the intelligence at the network components. H.323 places intelligence everywhere.

There are a variety of devices, protocols and configurations seen in typical VoIP deployments today. The components of VoIP include: end-user equipment, network components, call processors, gateways, optional elements, and protocols.

[*] M.Tech. Student, Kanwal Rekhi School of Information Technology, Indian Institute of Technology Bombay, Powai, Mumbai-400076. email: rahuls@it.iitb.ac.in

[†] Associate Professor, Kanwal Rekhi School of Information Technology, Indian Institute of Technology Bombay, Powai, Mumbai-400076. email: sahoo@it.iitb.ac.in

2.1 H.323

H.323 is a protocol suite that lays a foundation for IP based real-time communications including audio, video and data [WK05]. The architecture schematic is depicted in the figure 1.

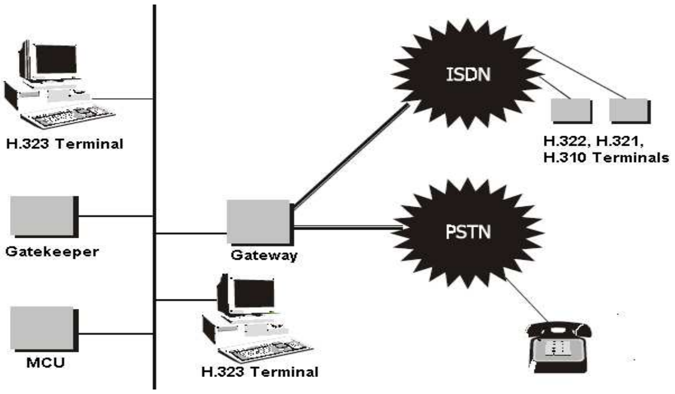
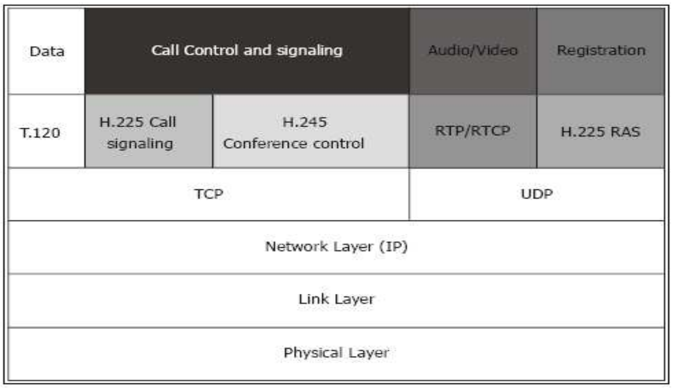


Figure 1: H.323 Architecture

The H.323 standard proposes an architecture that is composed of four logical components [Mit01] Terminals, Gateways, Gatekeepers and Multi-point Control Units (MCUs). *Terminals* are LAN client endpoints that are normally bound to a specific address and gateway, and provides real-time, two-way communication with either another H.323 terminal, an H.323 gateway or an MCU. *Gateway* is an endpoint on the network that provides for real-time, two-way communications between H.323 terminals on the IP network with other ITU terminals on a switch-based network like traditional public switched telephone network (PSTN), SIP network or to another H.323 gateway. The gateways handle different transmission formats. *Gatekeeper* is the central point for all the calls within its zone and provides services to the registered endpoints such as address translation, admissions control, call signaling, call authorization and authentication, call management, call routing, accounting, and bandwidth management. *MCU* acts as an endpoint on the network for providing capability for three or more terminals and gateways to participate in a multi-point conference. The MCU consists of a mandatory Multi-point Controller (MC) and an optional Multi-point Processor (MP). The MC's functions are to determine the common capabilities of conferencing terminals, using the H.245 protocol. The multiplexing of audio, video and data streams is handled by the MP under control of the MC.

A schematic description of the H.323 protocol stack is given in figure 2. The unreliable but low latency UDP is used to transport audio, video and registration packets. Whereas the reliable but slow TCP is used for data and control packets in call signaling. The T.120 protocol is used for data transfer. H.323 provides three control protocols H.225/Q.931 call signaling, H.225/RAS call signaling and H.245 Media control. The H.225/Q.931 is used for call signaling control. The H.225/RAS channel is used for establishing a call from the source to the receiving host. After the call is established, H.245 is finally used to negotiate the media streams. There are audio codecs (G.711, G.723.1, G.728, etc.) and video codecs (H.261, H.263) that encode and decode the audio and video data.



Data	Call Control and signaling		Audio/Video	Registration
T.120	H.225 Call signaling	H.245 Conference control	RTP/RTCP	H.225 RAS
TCP			UDP	
Network Layer (IP)				
Link Layer				
Physical Layer				

Figure 2: H.323 Protocol Stack

2.2 Session Initiation Protocol (SIP)

SIP is used for initiating, modifying, and terminating a two-way interactive user session that involves multimedia elements such as video, voice, instant messaging, online games, and virtual reality [RSC^{+}02]. SIP is used in association with its other IETF sister protocols like the SAP, SDP and MGCP (MEGACO) to provide a broader range of VoIP services. The SIP architecture is similar to HTTP (client-server protocol) architecture. It comprises *requests* that are sent from the SIP user client to the SIP Server. The Server processes the request and *responds* to the client. A request message, together with the associated response messages makes a *SIP transaction.*

SIP is an application-level protocol, i.e., it is decoupled from the protocol layer that it's transported across. Using TCP allows use of *secure sockets layer (SSL)/transport layer security (TLS)* providing more security whereas, UDP allows for faster, lower latency, connections. SIP depends on Session Description Protocol (SDP) for negotiation of session parameters such as codec identification and media. It supports user mobility through proxy servers and redirecting requests to the user's currently registered location.

The SIP architecture (figure 3) specifies two components: user agents and servers. A *SIP User Agent* is an end system acting on behalf of the user. The UA software contains client and server components. *User Agent Client (UAC)* is the user client portion, which is used to initiate a SIP request to the SIP servers or the UAS, whereas *User Agent Server (UAS)* is the user server portion that listens and responds to SIP requests. *SIP Servers* provide SIP call setup and services. *Registration Server* receives and authenticates registration requests from SIP users and updates their current location with itself. *Proxy Server* receives SIP requests and forwards them to the next-hop server, which has more information of the called party. *Redirect Server* resolves information for the UA client. On receipt of the SIP request, it determines the next-hop server and returns the address of the next-hop server to the client instead of forwarding the request to the next-hop server itself (as in the case of SIP proxy).

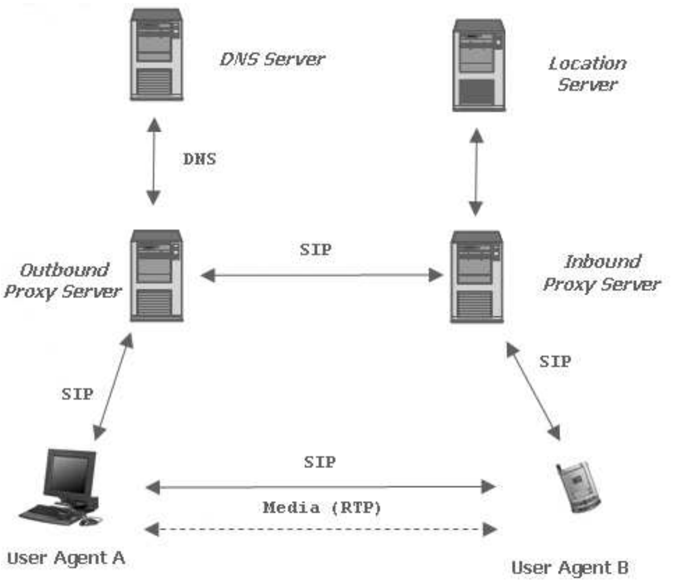


Figure 3: SIP Architecture

The endpoints begin by connecting with a proxy and/or redirect server which resolves the destination number into an IP address. It then returns that information to the originating endpoint which is responsible for transmitting the message directly to the destination. A security advantage of SIP is that it uses one port.

2.3 Media Gateway Control Protocol (MGCP)

MGCP exploded H.323's gatekeeper model and removed the signalling control from the gateway, putting it in a media gateway controller or soft-switch [RSC$^+$02]. This device would control multiple media gateways. A *Media Gateway* executes commands sent by the centralized *Media Gateway Controller (MGC)* and is designed to convert data between PSTN to IP, PSTN to ATM, ATM to IP, and also IP to IP, thus providing mechanisms to interconnect with other VoIP networks.

MGCP defines the communication between "Call Agents" (call control elements or MGCs) and gateways (figure 4). It is a control protocol that monitors the events on IP phones and gateways and instructs them to send media to specified addresses. These Call agents are assumed to have synchronized with each other and they issue coherent commands to the gateways under their control. The issued commands are executed by the gateways in a master/slave manner. MGCP defines the concepts of "Endpoints" and "Connections" to describe and establish voice paths between two participants. Similarly, it has defined "Events" and "Signals" to describe set-up or teardown of sessions.

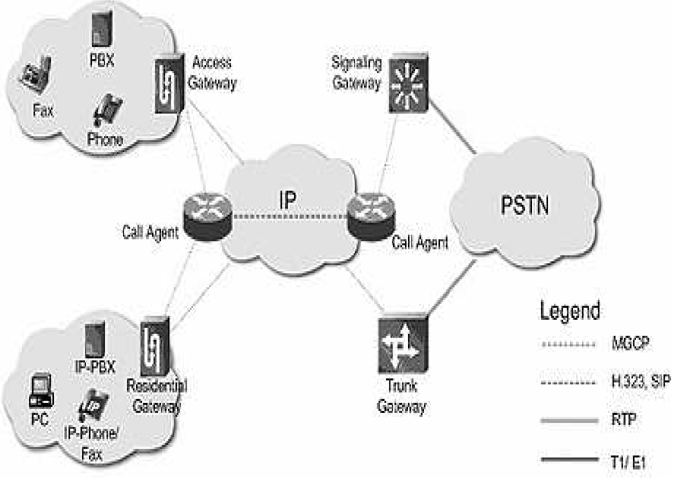


Figure 4: MGCP Architecture

3 VoIP Security Threat Scenarios

A VoIP deployment faces a variety of threats from different networking layers and areas of trust from within the network [Dha05]. For instance, an attacker can try to compromise a VoIP gateway, cause a denial-of-service attack to the Call Manager, exploit a vulnerability in a vendor's SIP protocol implementation or try to hijack VoIP calls through traditional TCP hijacking, UDP spoofing, or application manipulation. The attacks against a VoIP network can be categorized as follows:

- **VoIP Application Level Attacks** : At the application level, there are a variety of VoIP specific attacks that can be performed to disrupt or manipulate service. Some of them include:
  - **Call Hijacking** : An attacker can also spoof a SIP response, indicating to the caller that the called party has moved to a rogue SIP address, and hijack the call.
  - **Resource Exhaustion** : A potential DoS attack could starve the network of IP addresses by exhausting the IP addresses of a DHCP server in a VoIP network.
  - **Eavesdropping** : An attacker with local access to the VoIP LAN may sniff the network traffic and decipher the voice conversations. A tool named *VOMIT (voice over misconfigured Internet telephones)* can be downloaded to easily perform this attack.
  - **Message Integrity** : The attacker may be able to conduct a *man-in-the-middle* attack and alter the original communication between two parties.
  - **Toll Fraud** : An attacker can impersonate a valid user/IP phone and use the VoIP network for making free long distance calls.
  - **Denial of Service (DoS)** : DoS is caused by anything that prevents the service from being delivered. A DoS can be the result of unavailable

bandwidth or VoIP components being unavailable. Many things can cause a DoS including: a network getting congested to a level that it cannot provide the bandwidth needed to support the application; servers not capable of handling the traffic; extraneous services may be running that reduce the available resources to the server; malicious programs such as viruses and Trojan horses; other malicious programs with the purpose of causing DoS; or hacking activity.

By spoofing end-point identity, an attacker may cause a DoS in SIP-based VoIP networks by sending a "CANCEL" or "BYE" message to either of the communicating parties and end the call. Since SIP is UDP based, sending a spoofed ICMP "port unreachable" message to the calling party could also result in a DoS. If HTTP Authentication is being used, user-agents and proxy servers should challenge questionable requests with only a single 401 (Unauthorized) or 407 (Proxy Authentication Required), forgoing the normal response retransmission algorithm, and thus behaving statelessly towards unauthenticated requests. Retransmitting the 401 (Unauthorized) or 407 (Proxy Authentication Required) status response amplifies the problem of an attacker using a falsified header field value (such as Via) to direct traffic to a third party.

If DoS is caused by bandwidth constraints, potential solutions are increasing the bandwidth and/or isolating the VoIP traffic so that it gets service first. Various methods of ensuring servers don't stop working, such as fail-over methods like clustering, can help reduce DoS from failing components. Each component of the VoIP system offered by the vendor, should be evaluated, removing those that are unnecessary. Server size should be planned such that all desired vendor services and expected traffic can be supported, adding some percentage for expected growth. Defense against DoS attacks of public servers can best be done by locating the device with the public available IP addresses behind a firewall or other device that only allows communication from trusted sources. Also, harden the operating systems in use, removing all unnecessary services and applications from the servers and workstations, patching, etc.

- **Availability** : VoIP networks face a serious risk of availability. The availability risk result from availability-based attacks against protocols, endpoints, network servers, and the kind of attacks designed to reduce the quality of speech or that target simple equipment malfunction(s). The main risk, and one that is even more basic, is the lack of electricity to power endpoints and other elements making up an VoIP network or infrastructure. The VoIP infrastructure components interact with other computer systems on the IP network. They are thus more susceptible to a security breach than the equipment combining the PSTN, which is usually proprietary equipment whose operations are somewhat obscure. Any DoS attacks such as SYN floods or other traffic surge attacks that exhaust network resources (e.g. bandwidth, router connection table, etc.) could also severely impact all VoIP communications. Even worms or zombie hosts scanning for other vulnerable servers could cause unintentional traffic surges and crater availability of these VoIP services.

- **Physical Access** : Physical access to the network or to some network component(s) is usually regarded as an end-of-game scenario, a potential for total compromise in VoIP. For example, if a malicious party is able to gain unauthorized physical access to the wire connecting a subscriber's IP Phone to its network switch, the attacker will be able to place calls at the expense of the legitimate subscriber while continuing to let the subscriber place calls at the same time. With the PSTN, a similar scenario would unveil the malicious party when the legitimate subscriber took the handset off hook.

- **Non-Trusted Identities** : Without the proper network design and configuration of an IP telephony-based network, one cannot trust the identity of another call participant. The user's identity, the "call-ID" information (e.g. a phone number), can be easily spoofed. An identity-related attack might occur anywhere along the path signaling information is taking between call participants. A malicious party might perform digital impersonation, while spoofing an identity of a call participant or a targeted call participant, where the voice samples might have been gleaned from the IP telephony-based network itself.

- **Attacks against the underlying VoIP devices' Operating System** : VoIP devices such as IP phones, Call Manager, Gateways, and Proxy servers inherit the same vulnerabilities of the operating system or firmware they run on top of. For instance, the Cisco Call Manager is typically installed on Windows 2000 and the Avaya Call Manager on Linux. There are hundreds of remotely exploitable vulnerabilities in flavors of Windows and Linux operating systems for which there are numerous "point-and-shoot" exploits freely available for download on the Internet. No matter how secure an actual VoIP application happens to be, this becomes irrelevant if the underlying operating system is compromised.

- **The placement of Intelligence** : With the PSTN, the phones are no more than a "dumb terminal" where the telephony switch holds the actual intelligence. With VoIP signaling protocols, some or all of the intelligence is located at the endpoints. An endpoint supporting this type of signaling protocol will have the appropriate functionality and ability to interact with different VoIP components and services as well as different networking components within the VoIP network. A malicious party using such an endpoint, or a modified client, will have the same ability to interact with these components.

- **Configuration Weaknesses in VoIP devices** : Many of the VoIP devices in their default configuration may have a variety of exposed TCP and UDP ports. The default services running on the open ports may be vulnerable to DoS, buffer overflows or weak passwords, which may result in compromising the VoIP devices. For instance, multiple installations of the Cisco Call Manager that runs an IIS server were reportedly compromised by the Nimda and the Code Red worms.

- **Attacks against IP Infrastructure** : Compared to the PSTN, VoIP networks face a greater types of attacks, as a result of a combination of key factors outlined below [Tuc04]:

  – Since VoIP uses the IP protocol as the vassal for carrying both data and voice, it inherits the known (and unknown) security weaknesses that are associated with the IP protocol. For instance VoIP protocols rely on TCP and UDP as transport mediums and hence also vulnerable to any low level attacks on these protocols such as session hijacking (TCP), malicious IP Fragmentation, spoofing (UDP), TCP RST window brute forcing, or a variety of IP protocol anomalies which may cause unpredictable behavior in some VoIP services.

  – Although signaling and media might take different routes, they share the same medium: the IP network. Unlike the PSTN, where the only part of the telephony network both the signaling and media share is the connection between the subscriber's phone and its telephony switch (thereafter the signaling information will be carried on a different network physically separated from the media the SS7 network), with IP telephony no such isolation or physical separation between voice samples and signaling information is available, increasing the risk of misuse.

  – In several VoIP architectures, the signaling and media information traverses several IP networks controlled by different entities (e.g. Internet telephony, different service providers, different telecom companies). In some cases, it is not be possible to validate the level of security (and even trust) that different providers enforce with their network infrastructure, making those networks a potential risk factor and an attack venue.

- **VoIP Protocols Design Flaws** : IP telephony-related protocols were not designed with security as their first priority or as a prime design goal. Some of those protocols added security features when newer protocol versions were introduced. Other IP telephony protocols introduced some security mechanisms only after the IETF threatened not to accept a newer version of the protocol if security was not part of it. Despite such demands and an effort to introduce "decent" security mechanisms within some IP telephony protocols during their design phase, in some cases inappropriate security concepts were adopted only to satisfy the IETF. Some of those security mechanisms were simply not enough, regarded as useless or impractical, giving a false sense of security to the users of these IP telephony protocols.

  An example of a security technology that might cause more harm than good is encryption. Encryption affects voice quality since it adds delay on top of the usual delay experienced with an IP telephony-based network and therefore degrades voice quality. Although some IP telephony-related protocol specifications mandate the use of encryption, it is sometimes simply not feasible to use encryption with those protocols. An example is the draft version of the new RTP protocol, which mandates the use of *triple-DES (data encryption standard)* encryption. We need not forget that most IP Phones today are not powerful enough to handle encryption.

  The use of VPN technology is another good example of a security-related technology that degrades voice quality. Where we have more than two or three encrypted IP telephony "tunnels", voice quality is usually unbearable, the result of current encryption technologies combined with realtime multimedia demands.

  Another flaw, for example, is a signaling protocol that does not maintain knowledge about changes made to the media path during a call. If one is able to abuse the media path, the signaling path will remain unnotified and clueless about the changes performed to the media path. Another example is a signaling protocol that does not have an integrity-checking mechanism.

- **Improper IP Telephony network designs** : The current offered network designs for the implementation of IP telephony-based networks do not offer proper mechanisms to defeat several basic hazards to the IP telephony network. For example, IP telephony equipment (devices) is not authenticated to the network, and this makes the work of the phreaker easier; in some cases, by plugging a rogue device to the network, free phone calls can be made. Also in many IP telephony-based networks an IP Phone's (a user's) actual location is not checked against the credentials it uses. It is not enough that the network switch is able to perform "port security" and bind the port connected to an IP Phone with the phone's MAC address. There should be a mechanism to correlate between the credentials presented, the MAC address the phone is using, and the physical port on the network switch it is connected to.

- **Functional protocol testing or Fuzzing** : It is a method of finding bugs and vulnerabilities by creating different types of packets for that protocol which contain data, that pushes the protocol's specifications to the point of breaking them. These specially crafted anomalous packets are consequently sent to an application, operating system, or hardware device capable of processing that protocol, and the results are then monitored for any abnormal behavior (crash, resource consumption, etc.).

  Functional protocol testing has already led to a wide variety of DoS and buffer overflow vulnerability discoveries in vendor implementations of VoIP products that use H.323 and SIP. Many of these vulnerabilities have been the direct result of focused VoIP research conducted by the University of Finland's PROTOS group, which specializes in the security testing of protocol implementations. The PROTOS group typically makes their tools available to the public, which means anyone can download and run the tools necessary to crash vulnerable implementations.

4 H.323 Security Concerns

The four security goals, *authentication*, *integrity*, *privacy*, and *non-repudiation* are accomplished with the four mechanisms: *configuration*, *authentication*, *key exchange* and *encryption*. During the initial stage of configuration, the device is authorized to the network and may be authenticated. Integrity and privacy are accomplished through encryption using symmetric or asymmetric keys. A signature is attached to gain the fourth goal of non-repudiation [Wei01].

4.1 H.323 security : H.235

The *H.235* protocols of H.323 provide *privacy* (no eavesdropping), *message integrity* and *authentication* (ensuring that people really are who they claim to be) and are expressed as Annexes to H.235 Version 3 [KWF05]. They are Annexes D, E, F, G, H, and I as follows:

4.1.1 Annex D - Baseline Security Profile

In the baseline security profile, end-point and gatekeeper share a secret key which is used as basis for all cryptographic mechanisms [Tha05]. These keys are stored in the back-end service. On every end-point registration, the gatekeeper requests the shared secret key with that respective end-point from the back-end service. The gatekeeper uses this key to verify messages sent by the end-point (also the registration message) and to compute tokens for messages that it sends to the end-point. These tokens are values computed by an algorithm (e.g. Message Authenticated Code (MAC)) applied to the message together with the key. After the gatekeeper has calculated the token it appends it to the message and sends the message to the end-point. The end-point verifies the message that seems to come from its gatekeeper with its key and accepts it if the verification is successful.

The security provided by the baseline security profile works on a hop-by-hop basis (figure 5). A hop is a trusted H.235 element along the communication path (e.g. a gatekeeper, MCU, proxy). Hop-by-hop means that at every hop the security information is verified and recomputed. On a path containing two end-points and one gatekeeper there are two such hops. One hop from the first end-point to the gatekeeper and one hop from the gatekeeper to the second end-point. After the first hop, the gatekeeper verifies the authentication information from the first end-point, removes it from the message, adds authentication information for the second end-point to the message and forwards the message to the second end-point.

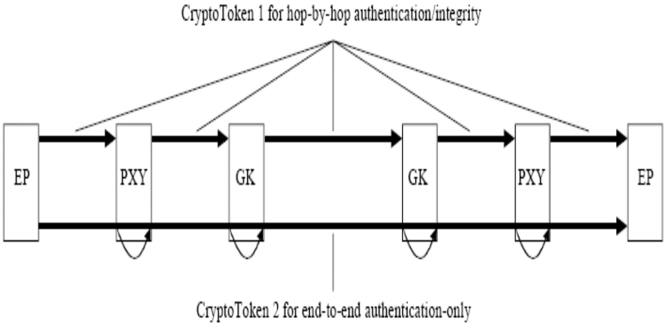

CryptoToken 1 for hop-by-hop authentication/integrity
EP
PXY
GK
GK
PXY
EP
CryptoToken 2 for end-to-end authentication-only

Figure 5: H235v3 Annex D - Simultaneous use of hop-by-hop security and end-to-end authentication

Table 1 shows the security services supported. *HMAC-SHA1-96* algorithm is used on the entire message which includes a monotonically increasing sequence number and timestamp. Then *CryptoH323Token* field is used to send this encrypted message to the next H.235 element. The gatekeeper upon receiving the encrypted message verifies the "authenticator" based on the liveness of the timestamp and the matching of the authenticator in the message with that computed by the gatekeeper. An "authentication only" option is available for smooth NAT/firewall traversal, so the integrity check is computed only over a special part of the message. Integrity protection of signalling data is optional.

The baseline security profile mandates the fast connection procedure. It does not prescribe confidentiality for call signaling. If desired, it may be implemented on a lower layer in the TCP/IP-stack. Confidentiality may be realized through other means such as IPSec or TLS. IPSec and TLS imply also authentication. The security features of H.235v3 concern the application layer. A disadvantage of this profile is the administration of all the shared secret keys. They have to be stored in a central place, which makes this one a critical part of the whole system.

4.1.2 Annex D - Voice Encryption Profile

The voice encryption profile handles media traffic security and may be combined with the baseline or the signature se-

Security Services	Call Functions		
	RAS	H.225	H.245
Authentication	Shared Secret (Password), HMAC-SHA1-96	Shared Secret (Password), HMAC-SHA1-96	Shared Secret (Password), HMAC-SHA1-96
Integrity	Shared Secret (Password), HMAC-SHA1-96	Shared Secret (Password), HMAC-SHA1-96	Shared Secret (Password), HMAC-SHA1-96
Key Management	Subscription-based password assignment	Subscription-based password assignment	

Table 1: H235v3 Annex D - Baseline Security Profile

Security Services	Call Functions		
	H.225	H.245	RTP
Confidentiality			56-bit DES or 56-RC2/ 168-bit Triple-DES, AES
Key Management	Authenticated Diffie-Hellman key agreement	Integrated H.235 session key management; certificate requests	

Table 2: H235v3 Annex D - Voice Encryption Option

curity profile. It describes the master key exchange during H.225 call signaling and the generation and distribution of media stream keys during H.245 call control. It is optional, because certain IP telephony environments already offer a certain degree of confidentiality (e.g. a dedicated telephony network operated on copper cables inside a building). However, it may be applied if additional media confidentiality is desired. Table 2 shows the security services supported by the voice encryption profile.

4.1.3 Annex E - Signature Security Profile

Signature Security Profile provides authentication, message integrity and non-repudiation using asymmetric methods like Digital Signatures on every message. The application of the GK-routed model (figure 6) and the fast connect procedure are mandatory. The Digital Signature model is an optional model in the standard. It introduces improved security through the use of Digital Signatures. Because there is no need of storing secret keys for all end-points, this model is also more scalable and better suited for a global VoIP solutions.

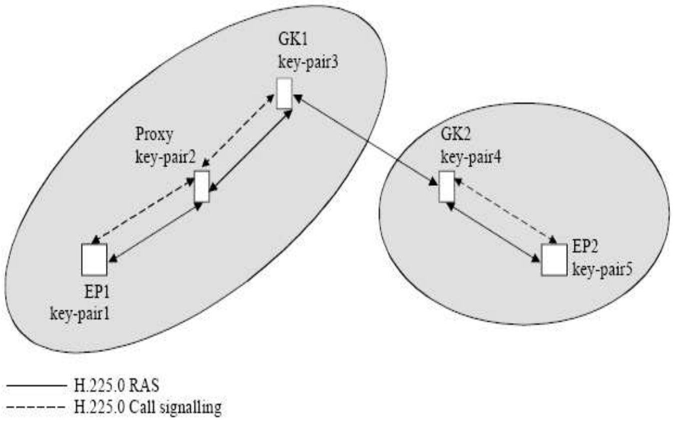


Figure 6: H235v3 Annex E - Illustration of public key usage in a gatekeeper routed model

Certificates in general give some assurance that the presenter of a certificate is who he claims to be. The intention is to authenticate the user not only the end-point. With the usage of digital certificates a user can prove that he possesses the private key corresponding to the public key contained in the certificates. A verification of the certificates diminishes the risk of *man-in-the-middle* attacks.

The protocol describes the messages necessary for exchanging the certificates. But it does not specify the criteria by which they are mutually verified and accepted. Certificate policies are neither prescribed in this recommendation. However, applications using this framework may impose high-level policy requirements. But neither how these policies could look like nor how they are implemented is part of the standard.

Table 3 shows the security services of the signature profile. As in the baseline security profile, the security services authentication and integrity are supported. However, because a Digital Signature is assigned to a particular person, the actual person sending a message can be determined. It cannot claim that it did not perform that action. This property is called *non-repudiation.* The baseline security profile does not support it. The allocation of the certificates is part of the public key infrastructure which is not part of the standard.

4.1.4 Annex F - Hybrid Security Profile

The Hybrid Security Profile is a hybrid of Annex D and Annex E, relying both on asymmetric and symmetric techniques. Certificates and Digital Signatures are used to provide authentication, non-repudiation and message integrity for the first handshake between two entities. During this handshake, a shared secret is established that will be used further on in the same way described for the Baseline Security Profile. The hybrid security profile mandates the gatekeeper-routed model. Since the profile relies on a public key infrastructure rather than on pre-established shared secrets, it scales for larger, global environments.

Security Services	Call Functions		
	RAS	H.225	H.245
Authentication	SHA1/ MD5, Digital Signature	SHA1/ MD5, Digital Signature	SHA1/ MD5, Digital Signature
Non-repudiation	SHA1/ MD5, Digital Signature	SHA1/ MD5, Digital Signature	SHA1/ MD5, Digital Signature
Integrity	SHA1/ MD5, Digital Signature	SHA1/ MD5, Digital Signature	SHA1/ MD5, Digital Signature
Key Management	Certificate Allocation	Certificate Allocation	

Table 3: H235v3 Annex E - Signature Security Profile

Security Services	Call Functions		
	RAS	H.225	H.245
Authentication	RSA Digital Signature or HMAC-SHA1-96	RSA Digital Signature or HMAC-SHA1-96	RSA Digital Signature or HMAC-SHA1-96
Non-repudiation	only for first handshake	only for first handshake	
Integrity	RSA Digital Signature or HMAC-SHA1-96	RSA Digital Signature or HMAC-SHA1-96	RSA Digital Signature or HMAC-SHA1-96
Key Management	Certificate Allocation or authenticated Diffie-Hellman key agreement	Certificate Allocation or authenticated Diffie-Hellman key agreement	

Table 4: H235v3 Annex F - Hybrid Security Profile

4.1.5 Annex G - SRTP and MIKEY usage

Annex G discusses the incorporation of key management supporting the *Secure Real-time Transport Protocol (SRTP)*. SRTP provides confidentiality, message authentication and replay protection to the RTP/RTCP traffic. SRTP may be used within multimedia sessions to ensure a secure media exchange.

SRTP does not define key management by itself. It rather uses a set of negotiated parameters from which session keys are derived. The preferred key management solution is *Multimedia Internet Keying (MIKEY)*. MIKEY describes a key management scheme that addresses real-time multimedia scenarios (e.g. SIP calls and SRTP sessions, streaming, unicast, groups, multicast). MIKEY defines three options for the user authentication and negotiation of the master keys, all as 2 way handshakes. They are symmetric key distribution, asymmetric key distribution and Diffie-Hellman key agreement protected by Digital Signatures.

Annex G discusses the use of MIKEY to integrate key management suitable for SRTP in three profiles:

- Profile 1 using symmetric techniques to protect the key management data in gatekeeper routed scenarios;
- Profile 2 using asymmetric techniques to protect the key management data in scenarios with a single gatekeeper instance;
- Profile 3 describes Profile 2 for multiple intermediate gatekeepers.

4.1.6 Annex H - RAS Key Management

This profile discusses key management negotiation during the RAS gatekeeper discovery phase. During gatekeeper discovery, a shared secret is established between the endpoint and the gatekeeper. The negotiation of the shared secret may be protected using PINs or passwords during the initial phase of the protocol.

4.1.7 Annex I - H.235 Annex D for Direct Routed Scenarios

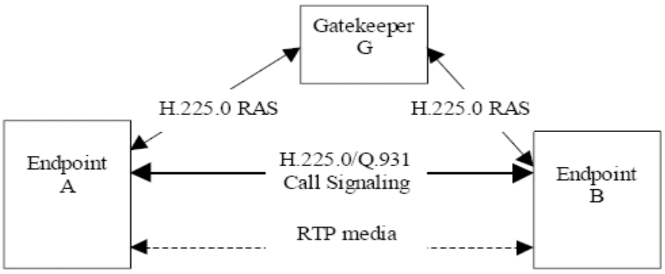


Figure 7: H235v3 Annex I - Direct-routed Call Scenario

Annex I enhances the Baseline Security Profile as well as the Hybrid Security Profile with the option to be applied in an environment where direct routed calls are performed using the gatekeeper for address resolution. The gatekeeper serves in this scenario as the key distribution center (KDC), issuing two tokens, one containing the key material secured with the caller's encryption key and the other one secured with the called party encryption key. The gatekeeper also generates a session key, which is applicable for the communication between the two endpoints involved in the call, and encrypts this key material using the previously derived encryption keys. The encrypted session keys are then transmitted back to the caller. The called party is sent this key as part of the SETUP message.

4.2 H.323 Security Methods

- **Authentication** can be done at call setup time, using TLS for instance, or while securing the H.245 channel. For authentication schemes which do not use certificates, H.235 allows challenge response exchange. When a certificate is used, H.235 does not describe the contents of the certificates, but provides ways to exchange certificates and verify the identities of the presenters.

- **Tokens** are generally parameters transmitted within H.323 messages that are opaque to H.323 itself but can be used by higher level protocols. H.235 uses two types of tokens:

  1. a *ClearToken* is an Abstract Syntax Notation number One (ASN-1) sequence of optional parameters such as time-stamp, password, Diffie-Hellman parameters, challenge, random number, certificate, generalID.
  2. a *CryptoToken* contains as object identifier of the encrypted token, followed by a crypto algorithm identifier, some parameters used by the algorithm (e.g. initialization vector), and the cryptographic data itself. CryptoTokens can be used to convey hidden tokens, signed tokens or hash values. The algorithm needs a key of a specific size N. For symmetric key algorithms the key is derived from a secret shared between the communicating parties.

- **Identity verification methods** : We suppose that terminals A and B have first exchanged their identities in clear form, then either *Symmetric Encryption* or *Hashing* or *Certificates and Signatures* can be used to verify the identities. In all methods, the timestamp prevents replay attacks.

- **Shared secret with Diffie-Hellman** : If the communicating endpoints do not share a secret, they must create a common one, using the Deffie-Hellman key negotiation [Res99].

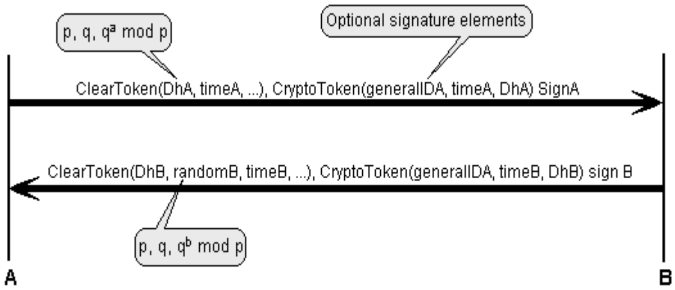

p, q, q^a mod p
Optional signature elements
ClearToken(DhA, timeA, ...), CryptoToken(generalIDA, timeA, DhA) SignA
ClearToken(DhB, randomB, timeB, ...), CryptoToken(generalIDA, timeB, DhB) sign B
p, q, q^b mod p
A
B

Figure 8: The Deffie-Hellman key negotiation

- **Securing RAS** : *RAS* messages are exchanged between an endpoint and a gatekeeper prior to any other communication. H.235 does not ensure privacy on the RAS link, but provides authentication. If there are no previous relationship and no shared secret between the gatekeeper and the endpoint, they need to negotiate one. For this purpose a Diffie-Hellman negotiation takes place during the GRQ, GCF (gatekeeper request, gatekeeper confirm) phase using a ClearToken to convey the DHset of parameters needed by Diffie-Hellman.

  After this, the gatekeeper and the endpoint share a common secret, which can be used to authenticate any subsequent RAS messages between them, in particular the *registration request (RRQ)* and *unregistration request (URQ)*. This is done by including in those messages a CryptoToken (encrypted using the DH secret) with the XORed combination of the *GatekeeperIdentifier*, sequence number and gatekeeper provided random value.

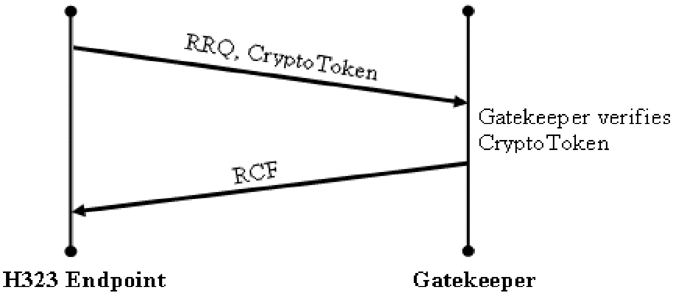


Figure 9: Securing RAS

- **Securing the call signalling channel (H.225)** can be done using TLS or IPSec. Authentication between the communicating terminals can be done at this stage. In the Setup, the caller will indicate which security schemes it supports for the H.245 channel.

- **Securing the call control channel (H.245)** is done using the method negotiated in the call signalling channel in the initial setup procedure. Different methods can be used to initiate the secure channel, depending on whether the communicating endpoints share a secret or not.

- **Securing the media channels** : Once the H.245 channel is secured, the terminals need to know which security modes can be used for the media channels. This is part of the *capabilities exchange*: terminals can signal that they support *GSM (global system for mobile communications)* capability and/or encrypted GSM capability.

  When a new logical channel is opened, the security mode is specified (chosen by the source) and the key that will be used for logical channel encryption is provided by the master either in the *OpenLogicalChannel* or in the *OpenLogicalChannelAck*. The key is associated with a dynamic payload type, so a receiver which has just been given a new key will know it must use it as soon as the payload type of the RTP packets, it receives matches the payload type associated with the key. The key can be refreshed afterwards. The key negotiation can be inherently secure using certificate exchange, or can be secured by securing first the H.245 channel.

  For multipoint communication, the secured H.245 channel is established with the MCU, and therefore the MCU must be trusted. New endpoints arriving in the conference can retrieve other endpoints' certificates., through the MCU for checking whether the endpoints actually own those certificates.

5 SIP Security Concerns

The main concerns for security of SIP are *confidentiality*, *message integrity*, *non-repudiation*, *authentication* and *privacy*. Rather than defining new security mechanisms specific to SIP, SIP uses those provided by *HyperText Transfer Protocol (HTTP)* and *Simple Mail Transfer Protocol (SMTP)*.

Signal confidentiality is best provided with full encryption, however, since some SIP message fields such as the Request-URI, Route, and Via must be read or modified by some proxies, possibly other methods must be used. If however, the proxy can be trusted, then encryption at the transport and/or network layers may be the best solution. Security at the transport and networking layers accomplishes full packet encryption using IPSec on a hop-by-hop basis. TLS had been used, but has been deprecated. Full encryption requires support of the encryption method at each end point.

5.1 Message Exchange Security

The SIP messages can be secured by encrypting them using a *media encryption key* [HGP01]. To protect this key, the *Session Description Protocol (SDP)* requests and replies must be encrypted. There are many other reasons for protecting the SIP messages, such as hiding the origin or destination of calls and the related information fields (subject, etc.). SIP messages can also be authenticated, which is useful not only to prevent call spoofing but also for accounting and billing.

SIP messages can be encrypted hop by hop, for instance using IPSec. SIP also describes an end-to-end encryption strategy based on a shared secret key between the sender and the receiver, or on a public key mechanism. If a common secret key is used, the receiver of the message is able to decrypt a message encrypted by the sender. If a public key scheme is used, the sender encrypts the message using the public key of the receiver. This encryption can be performed by the sender of the request or by an intermediary security proxy.

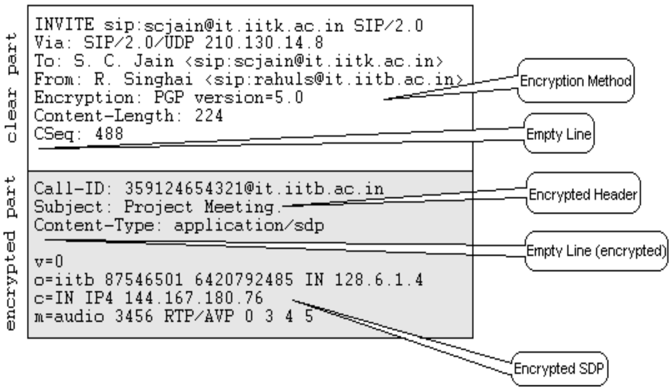
1. **Requests** : The request line and unencrypted headers are sent first and must include an Encryption header field that indicates the encryption method in use, e.g. *Encryption: PGP version=2.6.2, encoding=ascii.* The encrypted part begins after the first empty line. If only the message body has to be encrypted, an extra empty line must be inserted in the body before encryption to prevent the receiver from mixing up the message body data with encrypted headers.

2. **Replies** : The sender of the request should also indicate what key must be used to encrypt the reply. A SIP server receiving an encrypted request should not, in its reply, send in clear form of any field that was previously encrypted.

3. **Authentication** : SIP requests and replies can be authenticated using a Digital Signature. The Authorization header field is used for this purpose. It contains the signature of:

   - the first line (request line or status line)
   - all headers following the Authorization header
   - the message body.

   This semantic allows the exclusion of some variable fields (such as the Via field) from the signed data.

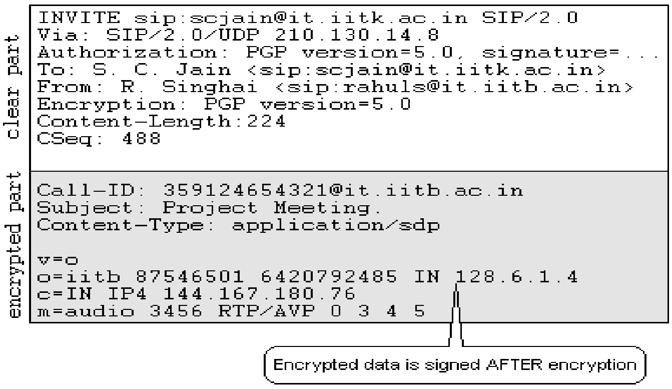


```
INVITE sip:scjain@it.iitk.ac.in SIP/2.0
Via: SIP/2.0/UDP 210.130.14.8
To: S. C. Jain <sip:scjain@it.iitk.ac.in>
From: R. Singhai <sip:rahuls@it.iitb.ac.in>
Encryption: PGP version=5.0
Content-Length: 224
CSeq: 488

Call-ID: 359124654321@it.iitb.ac.in
Subject: Project Meeting
Content-Type: application/sdp

v=0
o=iitb 87546501 6420792485 IN 128.6.1.4
c=IN IP4 144.167.180.76
m=audio 3456 RTP/AVP 0 3 4 5
```

Figure 10: Encryption in a request



```
INVITE sip:scjain@it.iitk.ac.in SIP/2.0
Via: SIP/2.0/UDP 210.130.14.8
Authorization: PGP version=5.0, signature=...
To: S. C. Jain <sip:scjain@it.iitk.ac.in>
From: R. Singhai <sip:rahuls@it.iitb.ac.in>
Encryption: PGP version=5.0
Content-Length:224
CSeq: 488

Call-ID: 359124654321@it.iitb.ac.in
Subject: Project Meeting.
Content-Type: application/sdp

v=o
o=iitb 87546501 6420792485 IN 128.6.1.4
c=IN IP4 144.167.180.76
m=audio 3456 RTP/AVP 0 3 4 5
```

Figure 11: Authentication through the Authorization header field

5.2 Transport and Network Layer Security

Transport or network layer security encrypts signaling traffic, guaranteeing message confidentiality and integrity [RSC^{+}02]. Two popular alternatives for providing security at the transport and network layer are, respectively, *TLS* and *IPSec*.

- *IPSec* is most commonly used in architectures in which a set of hosts or administrative domains have an existing trust relationship with one another. IPSec provides confidentiality and integrity for all traffic it receives from a particular interface. IPSec can also be used on a hop-by-hop basis. In many architectures IPSec does not require integration with SIP applications. User-agents that have a pre-shared keying relationship with their first-hop proxy server are good candidates to use IPSec.

- *TLS* provides transport-layer security over connection-oriented protocols; "tls" (signifying TLS over TCP) can be specified as the desired transport protocol within a *Via* header field value or a SIP-URI. TLS is most suited to architectures in which hop-by-hop security is required between hosts with no pre-existing trust association. TLS must be tightly coupled with a SIP application.

5.3 SIPS URI Scheme

The *SIPS URI scheme* adheres to the syntax of the SIP URI, although the scheme string is "sips" rather than "sip". The semantics of SIPS are very different from the SIP URI, however. SIPS allows resources to specify that they should be reached securely.

A SIPS URI can be used as an address-of-record for a particular user. When used as the Request-URI of a request, the SIPS scheme signifies that each hop over which the request is forwarded, until the request reaches the SIP entity responsible for the domain portion of the Request-URI, must be secured with TLS. Once it reaches the domain in question it is handled in accordance with local security and routing policy.

The use of SIPS in particular entails that mutual TLS authentication should be employed. Certificates received in the authentication process should be validated with root certificates held by the client; failure to validate a certificate should result in the failure of the request.

5.4 HTTP Authentication

SIP provides a stateless challenge-response-base mechanism for authentication, based on *HTTP authentication*, that relies on the 401 and 407 response codes as well as header fields for carrying challenges and credentials. When a proxy or user-agent receives a request, it may challenge to ensure the identity of the sender. Once identity has been confirmed the receiver should also verify that the requester is authorized to make the request in question. Each authentication is meaningful for a particular domain (or realm) and the realm string alone defines the protection domain.

Without significant modification, the reuse of the HTTP Digest authentication scheme in SIP allows for replay protection and one-way authentication only, without message integrity or confidentiality. The usage of "Basic" authentication where user-name and password are passed in clear text, has been deprecated due to its weak security. Servers must not accept credentials using the "Basic" authorization scheme, and servers also must not challenge with "Basic".

In SIP, a user-agent server uses the 401 (Unauthorized) response to challenge the identity of a user-agent client. Additionally, registrars and redirect servers may make use of 401 (Unauthorized) responses for authentication, but proxies must not. The proxies may use the 407 (Proxy Authentication Required) response.

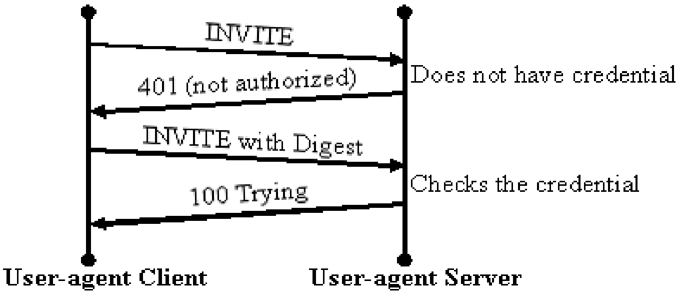


Figure 12: HTTP Digest Authentication Scheme

While a server can challenge most SIP requests, there are two requests that require special handling for authentication: ACK and CANCEL.

Under an authentication scheme that uses responses to carry values used to compute nonces (such as Digest), some problems come up for any requests that take no response, including ACK. For this reason, any credentials in the INVITE that were accepted by a server must be accepted by that server for the ACK. User-agent clients creating an ACK message will duplicate all of the Authorization and Proxy-Authorization header field values that appeared in the INVITE to which the ACK corresponds. Servers MUST NOT attempt to challenge an ACK.

Although the CANCEL method does take a response (a 2xx), servers must not attempt to challenge CANCEL requests since these requests cannot be resubmitted. Generally, a CANCEL request should be accepted by a server if it comes from the same hop that sent the request being canceled, provided that some sort of transport or network layer security association is in place.

5.5 Secure/Multipurpose Internet Mail Extension (S/MIME)

S/MIME is an enhancement to MIME that replaces *Pretty Good Privacy (PGP)* [DHRW98]. Since MIME bodies are carried by SIP, SIP may use S/MIME to enhance security. MIME contains components that can provide integrity and encryption for MIME data and S/MIME can be used for authentication, message integrity, non-repudiation of origin (using Digital Signatures), privacy and end-to-end data security (using encryption). It is also possible to use S/MIME to provide a form of integrity and confidentiality for SIP header fields through SIP message tunneling. S/MIME is particularly useful when full encryption of the packet is not feasible due to the need of network components to use data from the header fields.

S/MIME requires a public key infrastructure. Since certificates are associated with users, moving from one device to another may be difficult. The S/MIME certificates that are used to identify an end-user assert that the holder is identified by an end-user address. These certificates also contains the keys that are used to sign or encrypt bodies of SIP messages. Bodies are signed with the private key of the sender, but bodies are encrypted with the public key of the intended recipient.

6 Firewalls, Network Address Translation (NAT), and Call Establishment

Firewalls and NAT present a formidable challenge to VoIP implementers. All three major VoIP protocols, SIP, H.323, and H.248/MEGACO have similar problems with firewalls and NATs. Although the use of NATs may be reduced as IPv6 is adopted, they will remain a common component in networks for years to come, and IPv6 will not alleviate the need for firewalls, so VoIP systems must deal with the complexities of firewalls and NATs.

6.1 Firewalls and Network Address Translation (NAT)

A *firewall* is an in-line device or software component that monitors traffic between a trusted and untrusted network [Col05]. By being in-line, a firewall can use a rulebased policy to determine which packets are allowed to pass through and which are not. Firewalls work by blocking traffic deemed to be invasive, intrusive, or just plain malicious from flowing through them. By default, most firewalls deny all access. To explicitly allow certain types of access, rules are programmed into the firewall by the network administrator.

A useful property of a firewall, in VoIP context, is that it provides a central location for deploying security policies. This lends to the VoIP network where firewalls simplify security management by consolidating security measures at the firewall gateway, instead of requiring all the endpoints to maintain up to date security policies. This takes an enormous burden off the VoIP network infrastructure.

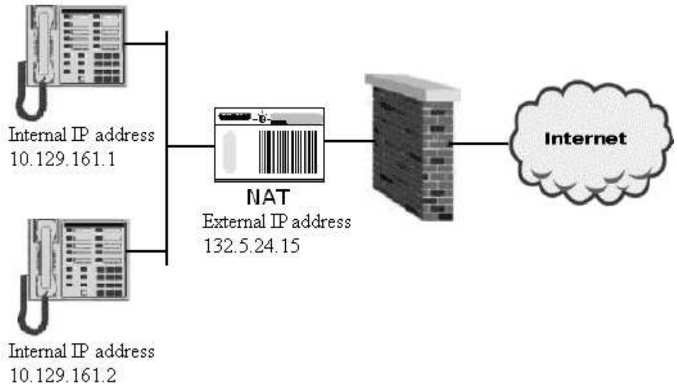


Figure 13: IP Telephones behind NAT and Firewall

Network Address Translation (NAT) was designed to preserve the limited IP space available with IP Version 4 (IPv4). NAT can be used to hide internal network addresses and ports, and enable several endpoints within a LAN to share the same (external) IP address.

NAT is a layer of security, making internal IP addresses less accessible from the public Internet. Thus, all attacks against the network must be focused at the NAT router itself. This provides security because only one point of access must be protected.

6.1.1 Firewalls, NATs, and VoIP Issues and Solutions

- Firewalls and NATs make it difficult for *incoming calls* to be received by a terminal behind the firewall or NAT. Allowing signal traffic through a firewall from an incoming call means leaving several ports open that might be exploited by attackers. Solution include careful administration and rule definitions if holes are to be punched in the firewall allowing incoming connections. Solutions without such holes, include Application Level Gateways and Firewall Control Proxies.

- NAT creates even more difficulties for incoming calls. Any IP application, that needs to make a connection from an external realm to a point behind a NAT router, would need to know this points external IP and port number assigned by the router.

- Firewalls and NATs affect *quality of service (QoS)* and can wreak havoc with the RTP stream. Both can degrade QoS in a VoIP system by introducing latency and jitter. They are also a bottleneck on the network because all traffic is routed through a single node. A solution to these issues is to use a VPN for all VoIP traffic.

- Firewalls have difficulties sorting through *VoIP signaling traffic.* RTP traffic is dynamically assigned an even port number in the range of UDP ports (1024-65534). The RTCP port controlling this stream also flows through an associated, randomly-assigned port. Allowing such traffic along such a vast number of ports by default would leave the system highly exposed. So firewalls must be made aware dynamically of which ports, media is flowing across and between which terminals. For this reason, only stateful firewalls that can process H.323 and SIP should be incorporated into the network to open and close ports. These firewalls can investigate application data in a packet. They open dynamic RTP ports (pinhole) by peeking into the H.323/SIP signaling messages. VPNs can also be used to tunnel through the firewall.

- The NATs have finite nature of NAT bindings. At a NAT, a public IP address is bound to a private one for a certain period of time (t). This entry gets deleted if no traffic was observed at the NAT for t seconds or the connection was torn down explicitly. A silence period during a conversation might be longer than t seconds. As a result, it is possible that some state information is destroyed before the call actually completes.

- NATs introduce major design complications into *media traffic control* in VoIP [Geo]. The assigning new port numbers at-random breaks the pair relationship of RTP and RTCP port numbers. The translation of IP Addresses and ports by NAT is also problematic for the reception of VoIP packets. The new addresses/ports may not correspond to those negotiated in the call setup process. Several available solutions for this include Realm-Specific IP (RSIP), IPv6 Tunnel Broker, IP Next Layer (IPNL), and UDP Encapsulation. Both RSIP and IPv6 Tunnel Broker need LAN upgrades. IPNL introduces a new layer between IP and TCP/UDP. UDP Encapsulation breaks down when both users are behind NATs.

6.1.2 Mechanisms to solve the NAT problem

- **Simple Traversal of UDP through NATs (STUN)** : STUN is a lightweight protocol that allows applications to discover the presence and types of NATs and firewalls between them and the public Internet, and to determine the public Internet Protocol (IP) addresses allocated to them by the NAT. STUN is a client-server protocol. A VoIP phone or software package may include a STUN client, which will send a request to a STUN server. The server then reports back to the STUN client what the public IP address of the NAT router is, and what port was opened by the NAT to allow incoming traffic back in to the network.

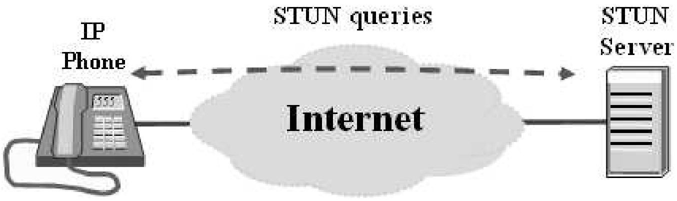


Figure 14: IP Telephones queries a STUN Server on the Internet

- **Simple Traversal of UDP Through NATs and TCP too (STUNT)** : STUNT extends STUN to include TCP functionality. It allows applications to determine external IP and port-binding properties, packet filtering rules and various timeouts associated with TCP connections through the NAT.

- **Traversal Using Relay NAT (TURN)** : TURN is a protocol that allows for an element behind a NAT or firewall to receive incoming data over TCP or UDP connections. TURN does not allow for users to run servers on well known ports if they are behind a NAT; it supports the connection of a user behind a NAT to only a single peer. The connection has to be requested by the TURN client. The TURN server would act as a data relay, receiving data on the address it provides to clients, and forwarding them to the clients.

- **Interactive Connectivity Establishment (ICE)** : ICE describes a methodology for NAT-Traversal for the SIP protocol and works through the mutual cooperation of both endpoints in a SIP dialog. In particular, it is used to allow SIP-based VoIP clients to successfully traverse the variety of firewalls that may exist between a remote user and a network.

- **Universal Plug and Play (UPnP)** : In UPnP the NAT is upgraded to support the UPnP protocol, and the client can query the NAT directly as to its external IP Address and Port number. However, UPnP does not scale to cascaded NATs, and there are potentially serious security issues with this solution, including vulnerability to denial of service attacks.

6.2 Call Setup Considerations with NATs and Firewalls

Many factors influence the setup time of a VoIP call. At the network level, these include the topology of the network and the location of both endpoints as well as the presence of a firewall or NAT. At the application level, the degree or lack of authentication and other data security measures, as well as the choice of protocol used to set up the call, can dramatically alter the time necessary to prepare a VoIP connection.

- **Application Level Gateway (ALG)** : A firewall with a VoIP ALG can parse and understand H.323 or SIP, and dynamically open and close the necessary ports. When NAT is employed, the ALG needs to open up the VoIP packets and reconfigure the header information therein to correspond to the correct internal IP addresses on the private network, or on the public network for outgoing traffic. This includes modifying the headers and message bodies (e.g., SDP) in H.323 and SIP. The drawback to ALGs is that they are embedded in the firewall itself, and thus the latency and throughput slowdown of all traffic traversing the firewall is aggregated and then compounded by the VoIP call volume.

- **Middlebox Solutions** : Middlebox-style solutions place an extra device outside the firewall that parses VoIP traffic and instructs the firewall to open or close ports based on the needs of the VoIP signaling via a midcom protocol (Figure 15). The drawbacks are that the firewall must be configured for control by the application-aware device, which incurs an initial setup cost, and the middlebox itself requires protection from attackers.

- **Session Border Controllers (SBC)** : SBCs are dedicated appliances that offer one or more of the following services to a VoIP perimeter: Firewall/NAT traversal, Call Admission Control, Service Level Agreement monitoring, support for lawful intercept, and protocol interworking.

7 Conclusion

VoIP technology is still at the early stage of adoption, and attacks against deployments have been largely un-

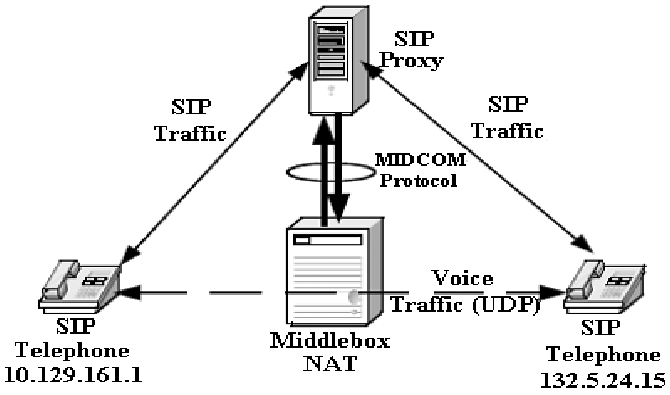


Figure 15: Middlebox Communications Scenario

heard of or undetected. As VoIP increases in popularity and numbers of consumers, so does the potential for harm from a cyber attack. Undoubtedly there are an abundance of vulnerabilities yet to be discovered in the implementations of other VoIP protocols such as H.245, H.235, H.248 through similar functional fuzzing techniques employed by the PROTOS group. It will be important to prevent these as-yet-undiscovered vulnerabilities from being exploited by enforcing selective conformance of VoIP protocols to their specifications.

We can expect to see more VoIP application-level attacks occur as attackers become savvier to the technology and gain easier access to test the VoIP infrastructure as it becomes more prevalent across residential areas. It will be important to keep track of calls, devices, users, and sessions to enforce security policy and prevent abuse of the VoIP network.

Security for a VoIP system should begin with solid security on the internal network. It should be protected from the threats of attached hostile networks and the threats of the internal network. The security policy should include any specific VoIP needs. The load of the VoIP system should be accommodated by the network and the servers involved, ensuring that proper resources are in place and available. Conducting a risk analysis of each component and process will identify the vulnerabilities and threats. This will provide the information needed to determine proper measures.

As we have seen that every security solution comes with a price, both in overhead and monetary terms. Therefore striking a balance between security and the business needs of the organization is key to the success of the VoIP deployment.

References

[Ark02] Ofir Arkin. Security Threats to IP Telephony-based Networks, December 2002.

[Col05] Mark Collier. Voice over IP and firewalls, February 2005.

[Dha05] Rohit Dhamankar. Intrusion Prevention: The Future of VoIP Security, June 2005.

[DHRW98] S. Dusse, P. Hoffman, B. Ramsdell, and J. Weinstein. S/MIME Version 2 Certificate Handling. RFC 2312 (Informational), March 1998.

[Geo] Adrian Geogescu. Best practices for SIP NAT traversal.

[HGP01] Olivier Hersent, David Gurle, and Jean-Pierre Petit. *IP Telephony*, volume 54. Addison Wesley Longman (Singapore) Pte. Ltd., 2001.

[KWF05] D. Richard Kuhn, Thomas J. Walsh, and Steffen Fries. Security considerations for voice over ip systems. *Compuer Security*, January 2005.

[Mit01] Debashish Mitra. Network Convergence and Voice over IP, March 2001.

[Res99] E. Rescorla. Diffie-Hellman Key Agreement Method. RFC 2631 (Proposed Standard), June 1999.

[RSC$^+$02] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler. SIP: Session Initiation Protocol. RFC 3261 (Proposed Standard), June 2002. Updated by RFCs 3265, 3853.

[Tha05] Johann Thalhammer. Security in voip - telephony systems. Master's thesis, Institute for Applied Information Processing and Communications, Graz University of Technology, Graz, Austria, 2005.

[Tuc04] Greg S. Tucker. Voice over Internet Protocol (VOIP) and Security. 2004.

[Wei01] Eric Weiss. Security Concerns with VoIP, August 2001.

[WK05] Thomas J. Walsh and D. Richard Kuhn. Challenges in securing voice over ip. *IEEE Security and Privacy*, 3:44–49, May 2005. ISSN:1540-7993.