Voice over IP Security: Threat Analysis and Countermeasures

Konstantinos Fysarakis

Supervisor: Dr. Allan Tomlinson

Submitted as part of the requirements for the award of the MSc in Information Security at Royal Holloway, University of London

September 2007

I declare that this assignment is all my own work and that I have acknowledged all quotations from the published or unpublished works of other people. I declare that I have also read the statements on plagiarism in Section 1 of the Regulations Governing Examination and Assessment Offenses and in accordance with it I submit this project report as my own work.

Signature

Date
Abstract

This Thesis describes the vulnerabilities of and threats to a VoIP network. Additionally, there is a detailed description of implementation techniques that will fortify the network, thus mitigating the risks identified, while maintaining quality of service.
Contents

1 Introduction .................................................. 3
  1.1 What is VoIP? ............................................ 3
  1.2 Why VoIP? .............................................. 5
  1.3 VoIP Security: Importance and Challenges ............... 11

2 VoIP - How it works ............................................. 15
  2.1 VoIP Call Control Protocols ............................. 15
    2.1.1 Session Initiation Protocol - SIP ..................... 15
    2.1.2 H.323 Protocol Family ................................ 17
    2.1.3 SIP vs. H.323 ....................................... 21
    2.1.4 Gateway Control Protocols ........................... 22
  2.2 Real-Time Protocol ...................................... 23

3 Preparatory Steps & Preliminary Attacks ....................... 26
  3.1 Footprinting ............................................. 27
    3.1.1 Website Research .................................... 28
    3.1.2 VoIP oriented Google Hacking ....................... 29
    3.1.3 WHOIS and DNS Analysis ............................. 31
  3.2 Scanning ................................................. 33
    3.2.1 Host/Device Discovery ............................... 33
    3.2.2 Port Scanning - Service Detection ................ 36
    3.2.3 Operating System Detection .......................... 37
  3.3 Enumeration ............................................. 39
    3.3.1 Banner Grabbing .................................... 39
    3.3.2 SIP Username Enumeration ........................... 40
    3.3.3 VoIP Support Services Enumeration ................. 41
    3.3.4 Enumerating VxWorks VoIP Devices ................. 42

4 VoIP Network Threats ........................................... 43
  4.1 Eavesdropping ........................................... 43
    4.1.1 Gaining Access to VoIP traffic ..................... 43
    4.1.2 Number Harvesting & Call Tracking ................ 44
    4.1.3 TFTP Configuration File Sniffing ................... 45
    4.1.4 Conversation Eavesdropping and Analysis ........ 45
    4.1.5 DTMF Decoding ..................................... 46
  4.2 Denial of Service Attacks ................................. 47
    4.2.1 VoIP Quality of Service ............................. 47
    4.2.2 Flooding Attacks .................................... 49
    4.2.3 Network Availability Attacks ....................... 51
Chapter 1

Introduction

1.1 What is VoIP?

The term Voice over Internet Protocol (VoIP) refers to the transmission of voice over packet-switched IP networks and was first introduced in the mid 1990’s. Since then its quality and reliability have improved steadily and, in conjunction with the increased network bandwidths and improvements in compression technologies, it has become a viable technology to the extent that it is now considered one of the most important emerging trends in telecommunications ([122], [75]).

VoIP can be used to make calls between two terminals, which can be two PCs, two VoIP phones, two conferencing units, two mobile phones or even two traditional phones (with part of the circuit between the two phones being an IP network). It can also be used to make calls between any combination of the aforementioned devices.

Other than the terminals, there has to be a way to deal with call setup, call control and data transport and, of course, the infrastructure (e.g. gateways) to allow communication through the various, possibly heterogeneous, networks that might separate the two terminals. Figure 1.1 shows a simplified VoIP network layout.

![Figure 1.1: Simplified VoIP network layout. ([34])](image_url)
The main applications for which VoIP technology may be deployed are private networks (e.g. a corporate network), carrier networks (e.g. replacing part of the PSTN backbone) and the Internet (e.g. through peer-to-peer softphones like Skype). Still, it should be noted that the term “Voice over IP” is typically associated with equipment on which users dial a phone number and communicate with another party who also uses a VoIP device or a traditional analog phone ([120]).

This Thesis will mainly focus on the security challenges of VoIP deployment in a corporate environment and to a lesser degree on Internet applications, while the security of VoIP technology at the carrier network level will not be covered.
1.2 Why VoIP?

Before you dial the number, listen until you hear a steady hum. This is the dial tone. Dial correctly. With your finger firmly in the hole, pull the dial around to the finger stop. Remove your finger and let the dial spin back. Do this until you have dialed the number. If you get a wrong number, say you are sorry, hang up. Check the number and dial again.

"We Learn About the Telephone", American Telephone & Telegraph Company, 1964

It has been a long time since 1876, when Alexander Graham Bell managed to transmit voice over a wire for the first time. The Public Switched Telephone Network (PSTN) has been evolving since then but, over 130 years later, it can no longer accommodate all the needs of its users and this is particularly evident from the rate at which VoIP is adopted.

Major telecommunication companies like British Telecom and Verizon provide VoIP services and also move to packet-switched networks themselves to be able to offer new services to their customers ([34]). Time Warner Cable, the second largest cable operator in the United States, offers "Digital Phone" services since 2003 ([7]). On the 15th of March 2007, the developers of Skype, the most popular VoIP softphones (software that emulates an IP phone), announced there were more than 171 million Skype users worldwide and that their software has been downloaded more than 500 million times. 73% of sampled Information Security Forum (ISF)\(^1\) members reported they are running evaluation VoIP projects, deploy the technology on a small scale or use it corporate-wide, restricting VoIP calls within the enterprise ([43]). Even in Greece, which has one of the lowest broadband penetration rates in the European Union (6.84% according to the latest official report from [42]), the convergence is starting to happen. Internet Service Providers team up with telecommunication companies - or get bought out by the latter - in order to offer "2 in 1" packages that include Voice and Data services over broadband internet connections, therefore eliminating the need for a traditional PSTN line. Moreover there are reports of "3 in 1" packages being available soon which will offer IP Television services in addition to Voice and Data. VoIP aware network devices (e.g. firewalls) are becoming a commonplace. In the consumer market, traditional PSTN phone manufacturers and even mobile phone manufactures are introducing new models with VoIP support. Two examples of such phones can be seen in 1.2 and 1.3.

\(^1\)URL: http://www.securityforum.org
Figure 1.2: Siemens Gigaset C450 IP: PSTN/VoIP Hybrid Phone

Figure 1.3: Nokia E65: Mobile Phone offering VoIP support via Softphone applications and integrated WLAN
It is clear that VoIP is indeed an important emerging trend in telecommunications and a technology that will probably change, in a few years from now, the way people communicate. Broadband internet connection speeds available to home users are improving day by day and the performance of Internet backbones - some of which, in the past, exhibited problems that affected their ability to support voice communication (see [81], [116]) - follows the same trend, making the Convergence an achievable goal. Figures 1.4, 1.5 and 1.6, taken from [123], demonstrate how networks have progressively moved from the physically segregated networks of the past, where each application had its dedicated network, to the converged networks of the future, where the segregation is only logical. Finally, Figure 1.7, taken from [113], shows an example of how a converged network could look like and the services it could potentially support.

![Figure 1.4: Traditional view - Data over PSTN using modems](image1)

![Figure 1.5: Current view - Dedicated network for data](image2)
Figure 1.6: Future view - High speed and reliable broadband access enables the Convergence

Figure 1.7: Example of a Converged Network
It is essential to examine both the limitations of PSTN and the potential of VoIP in order to fully understand what drives the expansion of VoIP and the convergence of voice and data in general.

Although the PSTN infrastructure was built to carry voice, data became the primary traffic on this infrastructure nowadays and not voice. However, data traffic has different characteristics to voice traffic (need for higher bandwidth, variable use of bandwidth, etc.) and therefore it cannot be transferred efficiently over circuit-switched networks.

It should be noted that for a PSTN call to take place, a 64kbps dedicated circuit between the two phones is utilized for the duration of the call. Telephone companies cannot use this bandwidth for any other purpose and so the two parties are billed for consuming its resources. In the case of packet-switched networks, bandwidth is used only when necessary. Moreover, compression can be utilized for an even more efficient use of the available bandwidth. Thus, generally speaking, VoIP phone bills are cheaper than PSTN phone bills and, in essence, VoIP eliminates "long-distance calls".

Data, Voice and Video cannot converge on the PSTN since most homes only have an analog line and 56kbps modem, which is the best PSTN can do in this case, does not offer adequate bandwidth. This is where packet-switched networks are clearly superior. With high-speed broadband access in every home all the aforementioned applications can run on top of the data-centric networks. In the PSTN backbone the convergence has already started ([24]).

PSTN is far from being an open infrastructure and has certain limitations that make the deployment of new technologies and features extremely slow. This is not acceptable in the modern era when users want more and more new features. It is also important for telecommunications companies to be able to offer these new features to their customers as soon as possible since competition in the open telecommunications market is very harsh. In PSTN the bearer channels, call-control and service logic are bound in one closed platform so it is not even possible to make changes to improve audio quality ([24]). This is not the case with VoIP, since its more open infrastructure allows vendors and carriers to launch new applications and features at will. The testimony ([7]) of John K. Billock (Vice Chairman and Chief Operating Officer of Time Warner Cable at the time) before the Federal Communications Commission (FCC) during the FCC VoIP Forum in 2003 is an excellent manifestation of the potential service providers see in the converged networks.

Additionally, recent research ([107]) has shown that VoIP increases the probability of cable operators' entry into new telecommunication markets since it significantly decreases the natural barriers to entry. The emergence of VoIP technology enables operators to offer telephone services in local markets where PSTN telephony could not provide log-run profit for them and allows competition to flourish in US local telecommunication markets.

In the corporate sector, a converged environment that offers voice and data over a single network is easier to manage and reconfigure, and helps reduce operating costs. The case of UBS Investment Research is just an example of a company that moved to VoIP, bypassing PSTN and which reported significantly reduced telecommunication costs ([56]). Typically, cost benefits from a move to VoIP reach savings of around 30% ([124]).

VoIP can also be seen as an enabler for doing business in a more productive and efficient way. Potential productivity enhancements through voice-based multimedia applications include:

- Internet-aware phones
- Real-time voice communication with the technical and sales support departments via a webpage.
• Interactive Voice Recognition (IVR); a technology which promises enhanced communication with customers and the ability to offer all kinds of services through the phone using fully automated voice recognition systems at the other end.

• Access to email, voicemail and fax through a single application.

• Inter-office trunking over Intranet.

• Remote connection to the corporate voice/data network for employees working at Branches or their home.

• Conference calls and call forwarding services without hardware changes

The above list is just a sample of the numerous applications and features that could be used on a VoIP-enabled corporate network. Any vendor can develop a similar applications or offer new creative solutions that can be installed on the existing network without the need for a hardware upgrade of the infrastructure.
1.3 VoIP Security: Importance and Challenges

As the use of VoIP expands to various critical applications, the importance of securing VoIP becomes apparent. At the Black Hat Security Conference in 2005 there was only one talk about VoIP security and a year later, in 2006, there was a whole session dedicated to the subject. VoIP made it to number six in the “Ten Most Important Security Trends of 2007” report ([40]) published by the SANS Institute. To quote the SANS report:

6. Voice over IP (VoIP) systems will be the target of cyber attacks. VoIP technology was deployed hastily without fully understanding security.

Additionally, more and more often, VoIP-related security incidents make it to the press. Quoting [61], as an example:

–VoIP Hacker Sentenced to Two Years
15 August 2007

A 23 year old man from Spokane, Washington, has been sentenced to two years in a federal prison and fined US$150,000 for his part in breaking into some of America’s largest IP telephony providers and defrauding them of more than a million dollars worth of call minutes. The man, Robert Moore, was able to break into various systems using simple dictionary and brute force attacks. Moore claims that “most of the telecom administrators were using the most basic password - Cisco, Cisco or admin, admin. They weren’t hardening their boxes at all.” His accomplice, Edwin Pena, has fled the United States after posting bail following his arrest.

The increasing importance of VoIP security has also led to the creation of organizations like the Voice over IP Security Alliance (VOIPSA)\textsuperscript{2} and numerous security websites and blogs dedicated to VoIP. All of the above as well as the fact that organizations such as the National Institute of Standards and Technology (NIST)\textsuperscript{3} issue security recommendations on the deployment of VoIP (e.g. [75]) and that security-oriented magazines dedicate whole sections to VoIP indicate that the Information Security community is aware of the dangers and there is research going on in the subject.

This interest in VoIP Security should not come as a surprise. The move to VoIP is generally considered inevitable but, if an organization is to rely entirely on VoIP for its communications, the confidentiality, integrity and availability of the service should not be at risk.

Just like data traffic, voice carries sensitive corporate information, if not the most sensitive ([82]), like discussions about a new deal or a new product. Having someone remotely switch on the speakerphones of IP phones at will, thus listening in on everything that goes on at the conference room or at the CEOs office, is clearly not an option. Additionally, sniffing traffic is probably easier than tapping a traditional phone line so specific mechanisms have to be utilized to provide a similar level of privacy. There has to be some sort of guarantee that any communication between two parties cannot be intercepted or modified and this should be the case regardless of the different forms of mobility supported by the IP Telephony, such as wireless connectivity ([37]).

The above are not the only things that can go wrong. Having customers who call the technical support line automatically redirected to the local bakery is not a risk any company would be willing to live with. A denial of service attack which disables the Phone Orders service during Christmas season is not a pleasant situation for any company either nor is having to pay ransom

\textsuperscript{2}URL: http://www.voipsa.org/

\textsuperscript{3}URL: http://www.nist.gov/
to some hacker group in order to get the VoIP infrastructure back online.

Reliability is another important issue. Traditional PSTN networks have reached a point of impressive reliability and a downtime of only a few minutes per year ([59]). On the other hand, data networks are nowhere near as reliable, with the exception of highly critical applications which justify a big investment into network redundancy. Taking into consideration that in a converged environment there is only one network and, if that network is lost, there is no fallback, it is easy to understand why regulators insist there is a PSTN for the routing of emergency calls ([69]). In May 2005, the Federal Communications Commission (FCC) adopted rules that require: "providers of interconnected VoIP services to supply 911 emergency calling capabilities to their customers as a mandatory feature of the service by November 28, 2005" ([41]).

Continuing on the issue of Regulatory Compliance, it is important to point out that existing regulatory requirements, such as the Sarbanes-Oxley Act of 20024 and the The Gramm-Leach Bliley Act of 19995 or even lawful interception laws such as the Regulation of Investigatory Powers Act 2000 (RIPA)6 and its United States equivalent, the Communications Assistance for Law Enforcement Act (CALEA)7, apply to corporate VoIP deployments as well ([83]). Consequently, getting the IP telephony security and auditing mechanisms right is a critical part of regulatory compliance for every organization.

When it comes to lawful interception provisions in specific, the situation can be considered a bit vague. In the United States, and as noted in [108], it is essential to distinguish between carrier and non-carrier use of IP telephony. In the case of non-carrier, like in a private corporate network, it seems that VoIP is considered an information service (just like e-mail for example) and thus CALEA is not applicable. In the case of carrier-like use or, in other words, in any case where a connection to the PSTN is present, the VoIP service is considered a telecom service for the purposes of CALEA. In fact, according to a statement8 of FCC issued in May 3, 2006, interconnected VoIP providers, thus anyone who provides VoIP services connected to the PSTN, were expected to comply with CALEA by May 14, 2007. For the time being though, the need for CALEA compliance by VoIP providers is tentative. Some VoIP companies and activists have filed a lawsuit9 against the FCC and, until the D.C. Circuit Court of Appeals issues a decision, nothing is certain ([74]). In the UK, on the other hand, things seem a bit more clear as RIPA’s definition of “public telecommunication systems” (for which provisions for lawful interception must be made), in addition to the carrier-like use of VoIP described above, includes private VoIP deployments on a private network connected to an Internet Service Provider (ISP) ([15]). Therefore, the rare cases of corporate VoIP deployments to which RIPA does not apply are those that do not feature any interconnection with the PSTN nor an ISP. Finally, legitimate private interceptions - for specific purposes such as monitoring for regulatory practices and adherence to standards, to prevent unauthorized use, to detect crime etc. - with the requirements and provisions described in RIPA, do apply to VoIP as well.

Quality of Service (QoS), as defined in [64], is: "The collective effect of service performance which determine the degree of satisfaction of a user of the service". QoS is another important consideration when dealing with IP telephony services. If VoIP is to replace PSTN it should be able to offer similar user satisfaction. Unfortunately, QoS in voice transmission over packet-

---

5URL: http://www.ftc.gov/privacy/privacyinitiatives/glbact.html
7URL: http://www.fcc.gov/calea/
9URL: http://www.eff.org/Privacy/Surveillance/CALEA/CALEARehearingFinalFull.pdf
switched networks is not something simple. For voice, unlike data applications, it is not sufficient to deliver the IP packets to the destination without errors. The packets must also arrive on time or with minimal variation in delay, so low latency is essential, and in the correct order to avoid jitter. Additionally, VoIP is extremely sensitive to packet loss. To make things more complicated, some of the security mechanisms required to fortify a VoIP deployment, like encryption, may add significant overhead, reduce network performance and have an impact on the QoS (see Figure 1.8). Situations where VoIP network administrators have to choose between providing security or fast network performance, are not uncommon ([84]). The subject of VoIP QoS will be discussed in greater detail later on.

![Figure 1.8: VoIP: Security vs. Quality of Service](http://www.voipsa.org/)

It is not uncommon for a VoIP deployment to cover multiple buildings, multiple cities or even multiple countries and support users that call from the office, their home or the road. This can make discerning the network resources that support the VoIP platform pretty hard. There are cases of large organizations where the IT staff have trouble identifying all their key network assets and the dependencies between. Identifying how a failure in each of those assets will affect the VoIP service is a much harder question to answer. Thus, securing the VoIP deployment and only parts of its supporting infrastructure is not an option. The importance of the security of the whole network infrastructure on top of which a VoIP system is built cannot be over-emphasized. An attacker will not waste time trying to brute force her way through the web interface of the voicemail system when the Linux server on which it is installed still uses the default password for the root account. If an attacker is able to locate and disable the TFTP server then VoIP phones which try to download configuration files on boot up will crash. If an attacker overwhelms the DHCP server then phones requesting an IP address on boot up will not be usable. These are just a few examples of how interwinded VoIP security is to the security of the underlying data network. The convergence of voice and data only magnifies the importance of good network security practice - like keeping every network device and host up-to-date with the latest patches.

To illustrate the complexity and the numerous challenges of VoIP security the main VoIP threat categories, as identified by the Voice over IP Security Alliance (VOIPSA)\textsuperscript{10} in its VoIP Threat Taxonomy project([38]), will be enumerated:

1. Social Threats

\textsuperscript{10}URL: http://www.voipsa.org/
2. Eavesdropping
3. Interception and Modification
4. Service Abuse
5. Intentional Interruption of Service
6. Other Interruptions of Service
Chapter 2

VoIP - How it works

A brief overview of the way VoIP works and the protocols involved is necessary in order to describe some of the advanced attacks that will be mentioned in later chapters. At this point it should be reminded that for Voice over IP transmission to take place there has to be a way to setup the call, to encode and to transfer data, gateways between the different networks (referred to as Media Gateways) and, of course, two or more terminals that will participate in the communication. The different types of devices that can act as terminals have already been mentioned.

2.1 VoIP Call Control Protocols

Call signaling (which refers to the setup, configuration, modification and termination of calls) is performed by a call processing manager or IP PBX. The most commonly used call control (or call signaling) protocols are \textit{H.323}, specified by the International Telecommunications Union (ITU)\textsuperscript{1} in 1996, and \textit{Session Initiation Protocol} (SIP), specified by the Internet Engineering Task Force (IETF)\textsuperscript{2} in 1999.

2.1.1 Session Initiation Protocol - SIP

SIP is a text-based protocol and its messages consist of Requests and Responses (like a challenge-response mechanism), some of which will be mentioned later in this section. A SIP Uniform Resource Indicator (URI) is used to address users. An example of a SIP URI, taken from RFC 3261 [49], follows:

\texttt{sip:alice:secretword@atlanta.com;transport=tcp}

The core components of a SIP deployment include User Agents (UAs) which are essentially terminals like IP phones and various servers like Registrar, Proxy, Location and Redirect servers which are responsible for user registration, routing of SIP requests as well as call-forwarding. Location, Registrar and Redirect services are often tendered by a single server. SIP is an application layer protocol and can be used with TCP, in which case SSL/TLS ([29]) can provide better security, or UDP, which allows for lower latency and faster connections.

Some common SIP Requests, mostly listed in RFCs 3261 [49] and 3265 [50], include:

- \texttt{INVITE Request} initiates a connection

\textsuperscript{1}URL: \url{http://www.itu.int}

\textsuperscript{2}URL: \url{http://www.ietf.org}
• **BYE Request** terminates an existing connection

• **OPTIONS Request** is used to agree on SIP messages and codecs that the other party understands

• **ACK Request** acknowledges a successful response

• **CANCEL Request** cancels an invite request (e.g. when there is no answer at the other end)

• **REGISTER Request** is sent from a phone to the SIP registrar in order to announce the IP address at which the specific username/phone number can be reached.

• **REFER Request** transfers calls and contacts external resources

• **SUBSCRIBE Request** sent to request to be notified if there is a change in a resource or call state

• **NOTIFY Request** provides information about a state change to the entities that have subscribed to be notified for such a change

SIP Responses (see [49]) are three-digit codes and the first digit indicates the category of the response, much like HTTP responses. Some examples of SIP responses would be: 180 **Ringing**, 200 **OK**, 401 **Unauthorized**, 503 **Service Unavailable**.

A typical SIP call flow and how the various requests are routed through the SIP architecture can be seen in figures 2.1 and 2.2.

![Figure 2.1: SIP Call Setup and Data Transfer. ([114])](image-url)
SIP offers three different challenge-response based authentication mechanisms: Basic Authentication, Digest Authentication and PGP Authentication ([122]). An overview of the SIP security mechanisms available in the various network layers can be seen in Figure 2.3, taken by [36].

2.1.2 H.323 Protocol Family

H.323 is an umbrella system for multimedia communications such as VoIP and video-conferencing. There are four main components in the H.323 architecture: two or more Terminals, a Gateway
for interfacing with other networks, a Multipoint Control Unit (MCU) which is necessary to offer conferencing services and an optional Gatekeeper for authentication and local call management. TCP or UDP is utilized, depending on the component of the protocol suite. Figure 2.4 illustrates the various components of H.323 and their transport mechanisms while Figure 2.5 shows the architecture of an H.323 network. Finally, Figure 2.6 shows an overview of a typical call realized using the aforementioned protocol family.

Figure 2.4: H.323 Protocol Family. ([114])
Figure 2.5: H.323 Architecture Overview. ([75])
Figure 2.6: H.323 Call Set-Up and Voice Data transfer. ([75])
H.323 allows the use of two types of authentication, through H.235: Symmetric Encryption based and Subscription based, the latter available in symmetric or asymmetric mode. Additionally, H.235 specifies the use of IPSEC, defined in [71], to handle authentication between the VoIP devices. ([122])

2.1.3 SIP vs. H.323

H.323 is a bit more complicated than SIP since it is actually a protocol suite that tries to cover all aspects of IP communications. Another important difference is that while SIP places intelligence at the endpoints of a communication, H.323 spreads intelligence in many parts of the network. Although H.323 was the first to come, SIP is becoming more and more popular and could be considered the dominant standard. This is because of its simpler nature but also because SIP is adopted as the protocol of choice for the - wired and wireless - Next Generation Networks by various standardization organizations such as the IETF\(^3\), ETSI\(^4\) and 3GPP\(^5\) ([45]). Moreover, most major enterprise VoIP vendors choose to integrate SIP into their products ([87]). Thus, this Thesis will mainly focus on SIP attacks - though the basic principles are the same regardless of the protocol targeted. A brief comparison of the two signaling protocols can be seen in Figure 2.7 while a more thorough comparison can be found in [95].

<table>
<thead>
<tr>
<th></th>
<th>H.323</th>
<th>SIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Architecture</td>
<td>Covers many services such as capability exchange, conference control, basic signaling and registration.</td>
<td>Covers only basic services such as basic call signaling, user location, and registration.</td>
</tr>
<tr>
<td>Message format</td>
<td>Binary in ASN.1 format, gives problems with firewalls.</td>
<td>Text-based, results in more bandwidth overhead and is easy to extend and debug.</td>
</tr>
<tr>
<td>Addressing</td>
<td>One address for each physical entity.</td>
<td>E-mail-like identifier for each user.</td>
</tr>
<tr>
<td>Complexity</td>
<td>High, due to the large number of protocols in the H.323 protocol stack.</td>
<td>Low.</td>
</tr>
<tr>
<td>Scalability</td>
<td>Poor since it was initially designed for LANs.</td>
<td>Good - designed for Wide Area Networks (WANs).</td>
</tr>
<tr>
<td>Delay</td>
<td>Possibility of high delay due to the complex signaling procedure.</td>
<td>Low delay, uses a simplified signaling procedure.</td>
</tr>
<tr>
<td>Security</td>
<td>Uses the security profiles defined in H.235.</td>
<td>Authenticated via HTTP mechanisms and can use any HTTP security features in the transport layer.</td>
</tr>
</tbody>
</table>

Figure 2.7: SIP versus H.323: A brief comparison. ([2])

---

\(^3\)URL: http://www.ietf.org/
\(^4\)URL: http://www.etsi.org/
\(^5\)URL: http://www.3gpp.org/
2.1.4 Gateway Control Protocols

A Media Gateway usually consists of a number of logical and physical components and is used to transmit data over heterogeneous networks. It converts VoIP data as required (PSTN to ATM, ATM to IP, PSTN to IP etc.) and also translates signaling protocols to the signaling protocol utilized in the destination network. For example, in the case of a VoIP to PSTN communication, the SIP signaling protocol must be translated the SS7 (specified by the International Telecommunications Union - Telephony Sector (ITU-T) in their Q.700 Series Recommendations [63]), of which a brief overview can be found in [21]. There are a number of control protocols which are used to manage Media Gateways such as the Media Gateway Control Protocol (MGCP - see [3]) or the ITU and IETF supported MEGACO (H.248 - defined in [53]) protocol and the proprietary Skinny Station Protocol from Cisco. The Gateway control protocols mentioned are, in essence, signaling protocols complementary to the ones mentioned above (SIP, H.323 etc.). These protocols are also necessary in cases of large-scale VoIP deployments where they are used to interconnect the various VoIP networks.

An overview of an MGCP deployment can be seen in figure 2.8.

![Figure 2.8: Media Gateway and Media Gateway Controller. ([75])](image)

MGCP recommends the use of IPSec for authentication and encryption. ([122])
2.2 Real-Time Protocol

As soon as a connection is established between two terminals, using any of the signaling protocol mentioned above, voice transmission must be initiated. The Real Time Protocol (RTP), documented in RFC 3550 [51], is used for this task. RTP is responsible for digitizing, compressing, potentially encrypting and, finally, packetizing the voice traffic and its packets are sent over the network using UDP datagrams. RTP provides timestamping, a sequence numbering, payload type identification and delivery monitoring but it does not provide any kind of timely packet delivery mechanism and instead relies on lower layer protocols for the task. Due to the connectionless nature of UDP, RTP cannot offer any assurances about the delivery or the order of packets. However, the sequence numbers present in an RTP packet allow applications running on the endpoints to check for lost or out of order packets.

RTP Control Protocol (RTCP) is part of RTP and may be used for quality control. If VoIP endpoints update RTCP information it may be possible, if packet loss is detected, to adjust the RTP transmission rate accordingly - but not all VoIP devices do so. More information on RTCP can be found in Chapter 6 of [51], where the control protocol is defined.

Figure 2.9 shows the header that RTP adds to each UDP packet.

![RTP Packet Format](image)

Figure 2.9: RTP Packet Format ([23])

The description of the RTP Header fields follows, taken directly from RFC 3550:

The first twelve octets are present in every RTP packet, while the list of CSRC identifiers is present only when inserted by a mixer. The fields have the following meaning:

version (V): 2 bits
This field identifies the version of RTP. The version defined by this specification is two (2).

padding (P): 1 bit
If the padding bit is set, the packet contains one or more additional padding octets at the end which are not part of the payload.

extension (X): 1 bit
If the extension bit is set, the fixed header MUST be followed by exactly one header extension, with a format defined in Section 5.3.1.

CSRC count (CC): 4 bits
The CSRC count contains the number of CSRC identifiers that follow the fixed header.

marker (M): 1 bit
The interpretation of the marker is defined by a profile. It is intended to allow significant events such as frame boundaries to be marked in the packet stream. A profile MAY define additional marker bits or specify that there is no marker bit by changing the number of bits in the payload type field (see Section 5.3).

payload type (PT): 7 bits
This field identifies the format of the RTP payload and determines its interpretation by the application. A receiver MUST ignore packets with payload types that it does not understand.

sequence number: 16 bits
The sequence number increments by one for each RTP data packet sent, and may be used by the receiver to detect packet loss and to restore packet sequence. The initial value of the sequence number SHOULD be random (unpredictable) to make known-plaintext attacks on encryption more difficult, even if the source itself does not encrypt according to the method in Section 9.1, because the packets may flow through a translator that does.

timestamp: 32 bits
The timestamp reflects the sampling instant of the first octet in the RTP data packet. The sampling instant MUST be derived from a clock that increments monotonically and linearly in time to allow synchronization and jitter calculations (see Section 6.4.1).

SSRC: 32 bits
The SSRC field identifies the synchronization source. This identifier SHOULD be chosen randomly, with the intent that no two synchronization sources within the same RTP session will have the same SSRC identifier. Although the probability of multiple sources choosing the same identifier is low, all RTP implementations must be prepared to detect and resolve collisions.

RTP audio is compressed and decompressed at the endpoints using a variety of codecs, such as G.711, G.729, G.723. The choice of codec depends on the application but G.711 is the most commonly used codec and it is the most tolerant to propagation delay (see [91]).

Since all VoIP product vendors utilize RTP for audio transmission, the protocol is an attractive target for attackers. The Sequence Number, Timestamp and SSRC offer some protection from packet injection because if those values are not the ones expected, the packets will be
ignored. This is not an obstacle if the attacker can successfully perform a Man-In-The-Middle (MITM) attack because in that case she will be able to predict the correct aforementioned values and include them in the malicious packets. Certain protocols offer encryption and authentication of the RTP media stream - but more on this later chapters.
Chapter 3

Preparatory Steps & Preliminary Attacks

The preparatory steps an attacker has to follow before launching an attack to the VoIP infrastructure of an organization are, to an extent, common to those of traditional network attacks. In this chapter we will examine how an external attacker would go about finding publicly available information about the organization that might assist her in the attack, scan the target network, pick up specific targets of interest and enumerate them before proceeding with advanced attacks described in later chapters.

The importance of these preparatory steps is generally not as significant in the case of an internal attacker. Such an attacker, e.g. a recently fired employee, can afford to skip most of the steps described below since she will most probably have access in the company’s internal network anyway. It should be noted though that most of the attacks described in the following chapter will be described from the perspective of an internal attacker or an external attacker who has compromised the internal network.

Internal attacks are very much a threat to an enterprise as indicated by recent security surveys like [47]. To make things worse there is a rise in the number of internal incidents reported. According to [32], in 2006, for the first time ever, the number of internal security incidents for large businesses exceeds that of external incidents.

This does not in any way mean that external attacks are not a considerable threat and that the countermeasures described in this chapter can be ignored. This especially true nowadays; in the era of social networking sites and databases of people, their personal details and their interests. With the amount of information available in public domain, anyone determined enough only needs the name of a person and soon an extremely sophisticated and well target social engineering attack can be launched ([26]). Moreover, it is safe to assume that an external attacker will, at some point, try to gain access to the internal network of her target and the safest way to succeed is through the steps described below.

This chapter cannot be considered a thorough analysis of the techniques a hacker might use to footprint, scan and enumerate a target network; that is beyond the scope of this Thesis. The subject is covered in detail on most good Network Security books like [104] and [86] and also on Penetration Testing courses like [73]. A more VoIP-oriented approach on the subject can be found in [36].

All of the aforementioned sources influenced the contents of this chapter.
3.1 Footprinting

This first step involves systematic profiling of the target organization using passive techniques and methods that utilize, among other things, the target’s website, Google, DNS and WHOIS records. The objective is to gather as much information about the organization’s VoIP deployment and network security as possible. In most cases there is plenty of highly sensitive information available in public domain which can significantly help an external attacker.

Footprinting is considered the most important part when trying to determine the security posture of the target, but it can also be the most arduous and a sound methodology is essential. As already mentioned in the introductory comments of this chapter, a detailed analysis of all the steps a thorough footprinting might include is beyond the scope of this Thesis. Figure 3.1, taken from [104], features a detailed list of technologies and critical information attackers can identify through a properly executed footprinting stage.

<table>
<thead>
<tr>
<th>Technology</th>
<th>Identifies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internet</td>
<td>Domain Name, Network blocks, Specific IP addresses of systems reachable via the Internet, TCP and UDP services running on each system identified, System architecture (for example, SPARC vs. X86), Access control mechanisms and related access control lists (ACLs), Intrusion detection systems (IDSes), System enumeration (user- and group names, system banners, routing tables, SNMP information)</td>
</tr>
<tr>
<td>Intranet</td>
<td>Networking protocols in use (for example, IP, IPX, DecNET, and so on), Internal domain names, Network blocks, Specific IP addresses of systems reachable via the intranet, TCP and UDP services running on each system identified, System architecture (for example, SPARC vs. X86), Access control mechanisms and related access control lists (ACLs), Intrusion detection systems, System enumeration (user- and group names, system banners, routing tables, SNMP information)</td>
</tr>
<tr>
<td>Remote</td>
<td>Analog/digital telephone numbers</td>
</tr>
<tr>
<td>Access</td>
<td>Remote system type, Authentication mechanisms</td>
</tr>
<tr>
<td>Extranet</td>
<td>Connection origination and destination, Type of connection, Access control mechanism</td>
</tr>
</tbody>
</table>

Figure 3.1: Footprinting - Why it is worth the attacker’s time.
3.1.1 Website Research

An up-to-date and informative World Wide Web presence is considered an essential part of a modern organization’s promotional and marketing efforts. Not only that but for some organizations their website is an important - and the only in some cases - way of doing business and communicating with their customers. Unfortunately, the wealth of information available in any proper website can become an important tool in the hands of an attacker with social engineering skills.

Some examples of potentially sensitive information usually present on a company’s website would be:

- **Organizational structure:** The names of people in important positions and the hierarchical structure of an organization can prove invaluable when trying to social engineer sensitive information out of employees. Such information is present in most corporate websites.

- **Corporate locations:** Again useful for social engineering attacks but also necessary when trying to understand the flow of data during a VoIP call. The location of the headquarters, branches etc. of a company also tell an attacker where she should go in order to sniff traffic data from the corporate wireless network - traffic which might include VoIP data in the case of a wireless VoIP deployment.

- **Job Listings:** It is not uncommon for the Human Resources (HR) department of an organization to publish job vacancies on the company’s website. These listings, when describing the skills required by the candidates, can contain highly sensitive information about the technologies used within an organization. A job listing which, among other things, states ”knowledgeable with Current VOIP, Cisco AVVID products including, Cisco Call Manager, Unity Voicemail, Cisco Voice Gateways and Cisco IPCC a plus.” certainly gives away a lot of useful information about the VoIP infrastructure of the company to a potential attacker.

- **Help/FAQ/Tech Support:** These sections of a website might include step-by-step setup guides or links to help files of VoIP equipment used within an organization. Such pages can help an attacker find sensitive information like the default PIN numbers of the voicemail service. Additionally an outsider can identify what brands and models of VoIP phones are used withing the company. These can be later looked up in vulnerability search engines like SecurityFocus\(^1\), where there is also information on how to exploit these vulnerabilities.

- **Phone numbers and Extensions:** By calling outside office hours, when it is unlikely that someone will answer the call, an attacker will probably hear a prerecorded voicemail message. Such a service is included in most VoIP systems. By simply listening to that message an attacker might be able to identify the VoIP system used since these prerecorded messages are vendor specific. For example, the open source Trixbox\(^2\) project will respond to a missed call by default with a female voice that says ”The person at extension X-X-X-X is unavailable. Please leave your message after the tone. When done, please hang up or press the pound key.[beep]”.

The previous examples should have made clear that some of the, otherwise benign, information available on a company’s website can become powerful weapons in the hands of a hacker who will connect the dots. Unfortunately, it is not easy to remove most of the information

\(^1\)URL: http://www.securityfocus.com/
\(^2\)URL: http://www.trixbot.org/
mentioned above from a website just to protect from footprinting. Such a measure would be unreasonable and would severely limit the services a company can offer through the Internet. What can be done is to ensure that there is no reference to default passwords and PIN numbers in the help pages and that the job descriptions listed in public locations do not mention any detailed technical information about the IT infrastructure of the company.

3.1.2 VoIP oriented Google Hacking

Today’s powerful search engines are an invaluable tool in the hands of anyone who uses the Internet. Trying to find specific information among the billions of pages that the World Wide Web consists of would be impossible if we did not have effective search engines like Google, Yahoo etc. at our disposal. The increasing effectiveness of the search engines available in WWW, and especially that of Google, has led to the development of a new hacking technique called Google Hacking. The term refers to the exploitation of the power and flexibility search engine algorithms currently demonstrate, for malicious causes. In this chapter we will focus on the manipulation of search engines for footprinting, and potentially hacking, a VoIP deployment. Using the advanced features of a service such as Google, an attacker can target several categories of search results that will assist him in compromising a VoIP network. Indicatively:

- **Vendor press releases and case studies:** When vendors sign a big sale deal with an important client they usually issue a press release to advertise their success. Additionally, when implementing a large scale project for a customer they might publicise a case study on their website where one can sometimes find information about the products used for the deployment of the VoIP platform. Finding this information is trivial. An attacker can find all the case studies published by Avaya.com simply by typing into Google:

  "site:avaya.com case study"

- **Resumes:** Much like job descriptions might disclose sensitive information about a company’s VoIP - and IT in general - infrastructure, resumes of ex-employees might be a good place to look for such information. An ex-employee of company X who took part in the deployment of their VoIP platform using Cisco gear will probably mention that project in his resume.

- **Forums/Mailing lists:** Technical mailing lists and forums are an invaluable channel through which people can share information, knowledge and experience. The danger lies in cases where IT staff, while trying to give enough information about the problem they face in order to ensure they will get help from their peers, disclose sensitive information about the equipment used in the company they work for. There have been reports ([73]) of network administrators disclosing IP addresses and names of internal machines, names of technical staff and even firewall configuration files on forums and mailing lists while trying to get help or help their peers.

- **Web-based VoIP logins:** Most VoIP devices provide a web-based management interface for remote administration and for users to change their personal settings like PIN numbers, voicemail and forwarding options. These web interfaces are not meant to be exposed on the Internet. There is danger of someone trying to brute-force the password and, even worse, of exposing a vulnerability in the underlying web server. Unfortunately, with a service like Google it is extremely easy to find such exposed web interfaces. For example, many Cisco CallManager installations feature a user options page at "/..ccmuser/logon.asp”. If a hacker knew that company "Target” was using a Cisco platform and she wanted to look for any exposed CallManager installations, she would type the following into Google:
To make things worse, some Cisco IP phones come with an administration/diagnostics web interface which reports some interesting information like non-password protected TFTP server addresses (see Figure 3.3). All that an attacker has to do in order to find such web interfaces of exposed devices is visit the Google webpage and type the following:

\texttt{inurl:"NetworkConfiguration" cisco}

The above examples should not give the impression that only Cisco devices are vulnerable to Google Hacking. All web-based VoIP phones and PBX’s that are exposed can be found by properly utilizing Google. Other examples would be the \textit{Asterisk IP PBX (intitle:asterisk.management.portal web-access)}, the \textit{Linksip Sipura Phones (intitle:"Sipura SPA Configuration")} and the \textit{Zultys Phones (intitle:"VoIP Phone Web Configuration Pages")}.

More Google VoIP hacking terms can be found at [35].

It should be noted that \textit{Snom Phones} (see Figure 3.2) feature a service called "PCAP Trace" which, assuming the phone is left with the default non-password protected settings, allows a hacker who Google-hacks to the web interface to capture live traffic. A potential exploitation of this "feature" can be especially dangerous if the phone is connected to a hub along with other phones.

When trying to protect an organization from getting "Google hacked”, one good measure is to try and use the very same tools the attackers would use, the advanced features of Google, to search for exposed systems. A security-conscious administrator could type search terms like the ones described above into Google, using 	exttt{"site:companyIworkFor.com"} as well in order to narrow the search down to the pages of the company she is working for. There is rarely a good
reason why a VoIP phone or PBX has to be exposed to the internet. If, for some reason, such a
device needs to be exposed to the WWW then it should be ensured that the default passwords
for the web logins have been changed. Additionally, there are services like Cyveillance\(^3\) that can
monitor and report the online public presence of an organization, including ”Google hacking”
exposure.

![Network Configuration](image)

**Figure 3.3: Google Hacking into a Cisco IP Phone CP-7960 Network Configuration Web Interface**

For further research on Google Hacking, not restricted to VoIP, \[78\] and \[79\] probably offer
the most extensive coverage of the subject. Another good, but not as extensive, source of
information is \[77\]. Additionally, there is a database of search terms related to various network
devices, the Google Hacking Database, which can be found at \[68\]. A similar database which
contains mostly VoIP-related search terms can be found at \[35\].

### 3.1.3 WHOIS and DNS Analysis

The Domain Name System (DNS) is a distributed database system used to map human-readable
hostnames to IP addresses. It also stores other information like the addresses of Mail Exchange
(MX) servers. Every organization with an online presence relies on DNS to route website traffic
and email messages to the correct hosts. More information on the DNS service can be found in
the relevant RFCs: \[89\], \[90\].

Additionally, five Regional Internet Registries (RIPs) exist that manage IP address allocations
and provide registration services.

The list of RIPs followed by the regions they serve:

\(^3\)URL: http://www.cyveillance.com/
Asia Pacific Network Information Centre (APNIC)\(^4\): Asia Pacific

African Network Information Center (AfriNIC)\(^5\): Africa

RIPE Network Coordination Centre (RIPE NCC)\(^6\): Europe, Middle East and parts of Asia and Africa

American Registry for Internet Numbers (ARIN)\(^7\): North and South America, part of Africa

Latin American and Caribbean IP Address Regional Registry (LACNIC)\(^8\): Latin America and the Caribbean

Most of the above Registries also feature a WHOIS search engine on their websites. This type of search reveals the IP address ranges an organization owns throughout their region. Alternatively, several websites offer a free WHOIS domain lookup service and will resolve the information regardless of country. Moreover, Unix-based operating systems support the whois command which can be used simply by typing "whois target.com" in the command line.

The IP addresses resolved by a WHOIS query provide a good starting point for scanning, which is mentioned in the next section of this chapter. After some WHOIS research a hacker can start to layout the external network topology. Using software tools like Solarwinds DNS Analyzer\(^9\) it is possible to graphically represent the DNS structure of the target, including SMTP servers found in the MX records. This way an external attacker can identify which servers are running DNS and SMTP services. Next, using a tool like Solarwinds DNS Audit\(^9\) the attacker can try a reverse DNS lookup for all the IP addresses owned by the target organization which will most probably return interesting DNS names mapped to some of those IP addresses. These could be host names such as voicemail.target.com, vpn.target.com and so on, which definitely provide good material for further probing.

The range of countermeasures against WHOIS and DNS analysis is limited since this type of information is meant to be public. However an organization can make sure that this information is presented in a way that makes it difficult for an external attacker to connect the dots. For example, registered administrative contact email addresses can be generic (e.g. webmaster@target.com instead of john@courier.mx.target.com) and servers can be named in a way that is less descriptive of their task (e.g. guesswhat.target.com instead of callmanager.target.com). Moreover, the organization’s DNS server should use Transaction Signatures (TSIGs) to allow only trusted hosts to perform zone transfers, otherwise hackers will be able to anonymously download the entire DNS database. One last countermeasure is to use anonymous DNS Service options - if supported by the hosting provider.

\(^{4}\)URL: http://www.apnic.net/
\(^{5}\)URL: http://www.afrinic.net/
\(^{6}\)URL: http://www.ripe.net/
\(^{7}\)URL: http://www.arin.net/
\(^{8}\)URL: http://www.lacnic.net/
\(^{9}\)URL: http://www.solarwinds.net/
3.2 Scanning

After footprinting the target organization an external attacker will have to employ active probing in order to identify reachable devices and hosts. Scanning involves techniques such as ping sweeps and port scans. The goal of this stage is to probe all the IP addresses identified in earlier stages in search of Internet-reachable systems and identification of the services running on each of those systems.

Again it should be reminded that a VoIP environment is not just phones, SIP proxies etc. Because the availability of a VoIP network relies heavily on supporting infrastructure, an attacker will not confine his efforts to devices and hosts running VoIP services. Other targets of interest include, but are not limited to, VPN servers, TFPT servers, DHCP servers, DNS servers and Firewalls. By the end of the scanning stage a hacker will probably be aware of any network-accessible VoIP and core network systems.

3.2.1 Host/Device Discovery

The first step is to find devices and hosts accessible on the target network. There is a number of tools, even highly automated ones, that promise to find all active systems on a network - not to mention there are whole books written on the subject. The reason is simple; network administrators rarely know everything that is connected on their network.

**ICMP Ping Sweeps**

Ping is a network tool included in operating systems that uses the ICMP protocol to discover if a target host is active. It sends ICMP ECHO REQUEST packets to an IP address and waits for a response in the form of an ICMP ECHO REPLY packet. Ping Sweep refers to pinging a range of IP addresses and/or network blocks and waiting for replies. This is probably the first technique an attacker will use since this is the easiest way to find active hosts.

Although ping sweeps can be executed using the standard ping command, this is not the most effective method. There are several free, easy to use and more effective tools for the job like **fping**. It is a Unix command-line tool that can send ICMP requests to multiple hosts in parallel (unlike most ping tools that wait for a response before pinging the next host on the list). Typing "fping -a -g 192.168.1.0/33" on the command line will launch a ping sweep at all IP addresses from 192.168.1.0 to 192.168.1.33 and will return the IP addresses of the hosts that responded.

A significantly more powerful and versatile Unix command-line scanning tool is **Nmap**. While Nmap can accomplish the same ping sweep as the fping one described above (using the "-sP" option), there are many more options and functionality that cannot be covered here. A good source of information on the use of Nmap is [44]. It should be noted that Nmap, if run from within the local subnet, will also reveal the Media Access Control (MAC) addresses of the targets. Each Network Interface Card (NIC) features a unique MAC address assigned by its manufacturer and this information can be extremely useful for an attacker since each MAC address is associated with an IP address using the Address Resolution Protocol (ARP). More on this later.

There is a number of host (and port) scanning tools for Windows as well and most feature a graphical user interface. Foundstone offers **Superscan** and Solarwinds offers **Ping Sweep**.

---

10 URL: http://fping.sourceforge.net/
11 URL: http://insecure.org/nmap/
12 URL: https://www.foundstone.com/resources/proddesc/superscan.htm
13 URL: http://www.solarwinds.net/
to name a few. There is also Nessus\(^{14}\), which is a fully featured vulnerability scanner that offers host and port scanning, among other things, and which runs on both Linux and Windows systems.

The success of the standard ICMP ping described above is far from guaranteed since many administrators block ICMP ECHO REQUEST packets at firewalls and routers for security reasons. However, other ICMP packet types (e.g. ICMP TIMESTAMP, ICMP INFORMATION REQUEST) may not be filtered. Several scanning tools can use non-standard ICMP messages for querying devices like SuperScan, which has already been mentioned, ICMPquery\(^{15}\), ICMPush\(^{16}\) and Icmpenum\(^{17}\).

Although ICMP traffic can be an invaluable tool in the hands of a network administrator, there is no reason why a router or a firewall should allow all types of ICMP packets, coming from external hosts, to enter the network. Most security devices, like firewalls, offer granular control over ICMP requests and responses, thus allowing applications that use ICMP packets to operate normally, without exposing the network to the security threat of accepting all ICMP traffic. When it comes to hosts, most good personal firewalls can be set to block ICMP traffic.

**ARP Pings**

The Address Resolution Protocol (ARP), defined in [98], “is a protocol that allows dynamic distribution of the information needed to build tables to translate an address A in protocol P’s address space into a 48-bit Ethernet address”. In essence, it is a protocol that allows the binding of the IP and Ethernet networking layers together. This is accomplished through the use of tables. Each entry of an ARP table maps the IP address allocated to a host to the MAC address of that host - which, as mentioned before, is unique to the specific NIC that host uses. Every time a host or device needs to communicate with another host or device at a specific IP address, ARP is used to determine the destination’s MAC address so that the two parties can communicate directly through Ethernet. The first host sends an ARP request with the IP address of the party it seeks to communicate with and waits for a reply from the second host, which will include its MAC address.

Compiling a table of IP addresses and their corresponding MAC addresses is extremely useful for man-in-the-middle and other attacks covered in later chapters, but this is not the only case where ARP comes in handy for hackers. It can also be exploited to bypass blocked ICMP rules on a local network. If a hacker, using a compromised host of the local LAN, sends an ARP broadcast frame to request MAC addresses through a range of IP addresses, she will be able to see which of these hosts are alive on the local network. This is called an ARP Ping and can be performed using a variety of tools, like Nmap mentioned earlier. Another command-line tool for ARP-pinging is ARPing\(^{18}\) and there is also the MAC Address Discovery tool\(^{19}\) which features a GUI.

There is not really much an organization’s network administrator can do to prevent ARP pinging, since ARP is a core component of the way Ethernet networks work. Still, The effectiveness of this probing technique can be limited by logically separating critical portions of the VoIP environment from the rest of the network using VLANs. As an additional measure, Intrusion Prevention Systems (IPSs) that detect excessive ARP broadcast requests should be utilized.

\(^{14}\text{URL: http://www.nessus.org/}\)
\(^{15}\text{URL: http://www.angio.net/security/}\)
\(^{16}\text{URL: http://packages.debian.org/unstable/net/icmpush}\)
\(^{17}\text{URL: http://www.nmrc.org/files/sunix/icmpenum-1.1.1.tgz}\)
\(^{18}\text{URL: http://freshmeat.net/projects/arping/}\)
\(^{19}\text{URL: http://www.solarwinds.net/}\)
TCP Ping Scans

TCP Ping scans are another way of detecting active hosts and devices in a network where all ingress ICMP traffic is being blocked. This technique exploits the TCP/IP connection flags and the mechanics of typical TCP/IP handshake. An attacker sends a TCP SYN (Synchronize sequence numbers) or TCP ACK (Significant acknowledgment field) packet to a commonly used TCP port on the target host and, if the target answers with an RST (Reset the connection), then she knows the host is alive. ACK packets are preferred by hackers since some stateless firewalls discard SYN packets in order to block new incoming connections.

Again, Nmap is the tool of choice for most hackers when launching a TCP Ping Scan and it can be customized to use either SYN or ACK packets. Hping3 is an alternative.

TCP pinging is relatively easy to deflect since many security devices like Firewalls, Intrusion Prevention (IPS) and Intrusion Detection Systems (IDS) as well as "smart" routers can detect such an activity and block it. Some devices drop the SYN or ACK packets to begin with while others block the offending host once a certain threshold of scanning traffic is reached.

SNMP Sweeps

Simple Network Management Protocol (SNMP) scanning is another, very effective, way of detecting active network hosts and devices. SNMP is an application layer protocol that enables online management of computers and networks. There are three versions of SNMP (details taken from [126]):

- Version 3 (1999), with more complete security features, specified in RFCs 2570-2576.

SNMP V.1 uses UDP and is thus a connectionless protocol. Later versions also support the use of TCP but for most practical purposes it is still run over UDP.

There are many security issues with SNMP, especially with the first version of the protocol. Version 1 and Version 2 offer weak Data Origin Authentication and Access Control services, since they rely on the community string for security which is nothing more than a cleartext password. Version 3 supports encryption but most of its advanced security features are not utilized for backward compatibility reasons.

More details about the SNMP protocol and its vulnerabilities can be found in Chapter 8 of [111]. An even more detailed analysis of SNMP, and Network Management protocols in general, can be found in [110].

Unfortunately, most VoIP phones utilize Version 1 of SNMP, again for backward compatibility reasons. To make things worse, most network administrators do not change the default community strings on their network devices and, on top of that, some VoIP phone vendors ship their phones with SNMP support but no option whatsoever to change the community string or, at least, turn off the service. It is, thus, extremely simple for a hacker to see a wealth of sensitive information by using simple SNMP clients, since lists of default SNMP community strings for popular devices can be found at various websites like [1].

\(^{20}\) URL: http://www.hping.org
Popular SNMP scanning tools include the windows-based SNMP Sweep\textsuperscript{21}, SNScan\textsuperscript{22} and several command-line tools for Unix-based operating systems like snmpwalk\textsuperscript{23}, Nomad\textsuperscript{24}, snmp-audit\textsuperscript{25}.

The most obvious way to protect an SNMP enabled VoIP network is to change the public and read/write community strings to something obscure, or at least less obvious than the default value. This will not offer enough protection from a determined attacker but it will at least limit the effectiveness of automated hacking tools which usually try the default community strings. Additionally, the network administrator should configure security devices to monitor the access to SNMP ports (typically UDP 161 and 162). Access Control Lists should limit access to these ports from authorized IP addresses only. Finally, if SNMP Version 3 is supported by the VoIP devices and hosts deployed, it should be preferred over the previous versions of the protocol. The aim should be to have all network device and network management traffic routed in an encrypted form through the network and, if possible, on its own Management VLAN. [115]

\subsection*{3.2.2 Port Scanning - Service Detection}

By employing the techniques described in previous sections, a hacker will manage to accumulate a list of active IP addresses. She will then, almost undoubtedly, further investigate each one of these live hosts using \textit{Port Scanning}. This involves connecting to TCP and UDP ports of the target hoping to find active services. It is essential for an attacker to know what services run on the target host or device because only then can she determine what vulnerabilities may be present. Even if there are no vulnerabilities to exploit on the target machine, the attacker may be able to interact with the application associated with an identified active service to accumulate further information about the VoIP deployment.

VoIP services typically run over TCP and UDP. The commonly used SIP protocol, for example, is usually implemented as a service on Phones and PBXs that listen on UDP and/or TCP port 5060. Other TCP services, not restricted to VoIP, include WWW, FTP and SMTP, while DNS, SNMP and DHCP are common UDP services.

The tool of choice for \textit{Port Scanning} and \textit{Service Detection} is, once more, Nmap. By properly utilizing the flags and options supported by Nmap, a hacker can find open ports on the live target and identify the services running on those ports.

One of the most effective port scan types is the TCP SYN scan discussed earlier. During a TCP SYN scan, which was discussed earlier in this chapter, Nmap sends a TCP SYN packet to a specific port pretending it wants to initialize a TCP connection with the target. If the target responds with a SYN/ACK packet the port is open, an RST packet means the port is closed and if there is no response the port is reported as "filtered". Using the "-sV" option of Nmap the attacker enables service detection which gives more information about the host, like its operating system, and the type of services it is running. For example, when scanning a Cisco Callmanager system which employs Cisco’s proprietary Skinny Call Control Protocol (SCCP) protocol the attacker will probably see a response on ports 2000 (callbook service), 2001 (dc service and 2002 (globe service).

Another effective type of port scanning is UDP scan, during which Nmap sends an empty UDP header to each UDP port of the target. If there is any response, in the form of a UDP

\textsuperscript{21}URL: http://www.solarwinds.net/
\textsuperscript{22}URL: http://www.foundstone.com/us/resources/proddesc/sns阎.htm
\textsuperscript{23}URL: http://net-snmp.sourceforge.net/docs/man/snmpwalk.html
\textsuperscript{24}URL: http://netmon.ncl.ac.uk/
\textsuperscript{25}URL: http://freshmeat.net/projects/snmp-audit/
packet, then there is a service listening to that port, otherwise the port is not used or filtered. In the case of a SIP Asterisk server a default Nmap scan will not reveal any VoIP services running. If the attacker switches to a UDP scan she will most probably spot an open UDP 5060 port (SIP). Still, this information is not enough to identify the exact type of the VoIP device and further investigation will be required on the part of the hacker and this will be the subject of the next section in this chapter. Additionally, the UDP scan might reveal more interesting UDP services running on the server like DCHP and TFTP and, again, this will come in handy for the hacker later on.

Using a non-Internet addressable IP address scheme for the internal network is considered good security practice and will prevent many types of probing coming from the Internet, but still for a hacker it might be trivial to get internal access to the target organization’s network. As already stated, it is important, in order to prevent internal scanning of the network, to have proper firewall rules and logical separation of the network using VLANs. Intrusion Prevention Systems and Stateful Firewalls should be employed, were possible, to detect port scans and block the offending host. It should be noted that doing so for UDP scans can backfire and should be avoided since UDP is a connectionless protocol and the source can be easily spoofed. By exploiting this security measure a malicious user might be able to perform a Denial of Service (DoS) attack. When it comes to hosts, proper configuration of the firewalls and disabling unnecessary services is the best a network administrator can do to protect against scanning.

### 3.2.3 Operating System Detection

The last step of the scanning phase is one that can potentially offer the attacker an even better classification of the live systems she has detected in the previous steps. Some more details about each host/device like the Operating System, Device type and firmware version used can certainly be dangerous in the hands of a malicious person. This type of information can help in determining the vulnerabilities present in the target hosts, it can also help to better tailor the exploits that will be used to a specific host and even offer good material for highly targeted social engineering attacks (e.g. "Hello, this is XXX XXX and I am calling from Cisco Tech Support to inform you about the newer firmware version for the CP-7960 IP Phones your organization is using.").

**Stack Fingerprinting**

Although we have already seen from previous scan types that some ports might give away information about the operating systems in use, an attacker needs to cross-check that information. This is the purpose of Stack Fingerprinting. The term refers to an advanced scanning technique during which a network security tool, like Nmap, examines the way target machines respond to a selection of TCP/IP probes. Consequently the results are matched against databases of known responses and thus the target machines are classified.

Nmap is the most commonly used utility for OS detection. It features stack fingerprinting that can be activated using the "-O" flag on the command line. More information about OS Detection via TCP/IP Fingerprinting and how this is implemented in Nmap can be found in [60]. Also, [46] offers a brief but comprehensive report on the subject.

Blocking various types of scanning, as mentioned earlier, can make things harder for the attackers. Unfortunately though, when it comes to stack fingerprinting, the detection techniques available are so many that, in essence, there is no easy way to prevent attackers from getting the information they want. The best way to limit information leakage about a VoIP deployment
is to disable unnecessary services and devices.
3.3 Enumeration

In the case of a poorly protected VoIP network - or IT infrastructure in general - it is not unlikely for a hacker to discover easy ways of compromising the Confidentiality, Security or Availability of the network just by going through the steps described in the previous sections. If this is not the case, a hacker will have to employ more intrusive techniques like active connections to interesting systems, in order to further expand on the information gathered during Footprinting and Scanning. This next step, called Enumeration, is aimed at identifying valid user accounts and user groups, applications, network resources, poorly protected resource shares and other sensitive information that might assist in further attacks.

3.3.1 Banner Grabbing

Banner Grabbing consists of connecting to remote services and applications and observing the output. This way an attacker can learn useful things about the target like the specific type of service that is running on it, what software is being used along with version numbers and so on. Banner Grabbing is an easy and, most of the times, effective way to inventory a target’s VoIP applications and hardware. By identifying such sensitive information about a specific active device or host the attacker can then concentrate on platform-specific techniques and known exploit routines. It can be considered a technique that a hackers will almost certainly try against a VoIP deployment.

Telnet is a command-line tool that comes with most Operating Systems and which can be used for Banner Grabbing. A more advanced tool for the task is Netcat\(^\text{26}\). Assuming a hacker has identified, through her Nmap scans described above, an active host at IP address \(xxx.xxx.xxx.xxx\) which has an open TCP port at 5061, she would target \texttt{netcat} to that IP and port. Then, using the appropriate input (since, judging by the port number and Nmap output, she could safely assume this is a SIP service), it is almost certain she would get the application listening to that port and the version of it as feedback. What the hacker can then do is visit an online vulnerability database (like SecurityFocus\(^\text{27}\)) and search for a known vulnerability for the specific version of the application. If there is indeed such a vulnerability and an exploit published for it, or is at least easy to craft, the system will be compromised. The importance of keeping every part of the VoIP deployment, as well as every part of the supporting network infrastructure, patched and up-to-date is obvious.

It should be noted that there is a number of Vulnerability Scanners available which automate almost every Scanning and Enumerating technique described up to now, including Banner Grabbing as well as the search in databases for known vulnerabilities for the specific applications and versions detected. One such Vulnerability Scanner is Nessus\(^\text{26}\) that was mentioned earlier and which is a popular open-source tool for hackers and security-conscious network administrators alike. VoIPaudit\(^\text{28}\) should also be mentioned since it is a commercial VoIP-specific vulnerability scanner and penetration testing tool.

There a few countermeasures a network administrator can apply to prevent banner grabbing, like edit the source code of open-source applications such as Asterisk or Apache to remove their banners or restrict access to administrative services to specific IP addresses only but, at the end of the day, a determined hacker will find a way to get the information she wants. It all comes down to keeping all systems up-to-date and patched and disabling all but the necessary services (e.g. a PBX server should not have the telnet service running since it will not be used).

\(^{26}\)URL: \url{http://netcat.sourceforge.net/}

\(^{27}\)URL: \url{http://www.securityfocus.com/}

\(^{28}\)URL: \url{http://www.voipshield.com/products/voipaudit.html}
3.3.2 SIP Username Enumeration

Valid usernames or SIP phone extensions as well as registration and proxy servers are necessary information if an attacker plans to launch some of the advanced attacks described later on. Wardialing every possible phone extension is one option but it is extremely time consuming and, of course, highly intrusive. The attacker can use various faster and more sophisticated enumeration techniques to get hold of this information. These methods rely on examining the error messages returned after sending REGISTER, OPTIONS and INVITE SIP Requests to servers and phones. The legitimate function of the aforementioned Requests is already covered in Chapter 2. All of the above methods start by sending a Request which includes a non-existent username so that the hacker can first observe the response to an invalid request which can then be used as a point of reference against which all other responses will be compared.

It is important to note that the techniques described below are not equally effective against all types of phones and servers. If, for a given pair of enumerating method and server/phone, the response to an invalid username is identical to the one that a valid but unauthorized username gets back, then this method cannot be used. The effectiveness varies according to the way each vendor has implemented the SIP protocol.

**REGISTER Enumeration**

By sending a REGISTER Request that contains a username she is certain is invalid, a hacker will watch how the SIP Server/Proxy responds and she will be able to tell if the specific server is vulnerable to REGISTER Enumeration. If this type of enumeration is indeed effective, then the server should send back a 403 (Forbidden) SIP Response, which means that the username does not exist. This is good news for the hacker since she can then go on trying possible usernames until she gets a 401 (Unauthorized) SIP Response, which would mean that the username does exist but authorization is required before registering. In the case that the server responds with a 401 to the original request then the server cannot be enumerated using this method since it gives an "Unauthorized" response even if the username does not exist.

**INVITE Enumeration**

INVITE Enumeration involves sending INVITE Requests to the registrar of the VoIP deployment and determining, based on the error messages sent as a response, which usernames are valid and which not. Again, the effectiveness of this method depends on the type of the SIP Server/Proxy used. Some proxy servers send a 503 (Service Unavailable) SIP Response in both the case of an invalid username and the case where the username is valid but no one picks up the phone, thus rendering this method ineffective.

*INVITE scanning* a VoIP deployment is not an option in some cases since it is way more intrusive than the other methods described in this section; it involves actually calling the various phones on the network (if the username that the hacker tried exists, of course). Even if calling after office hours, it will leave an audit trail both on the phones called and on the SIP proxy handling the calls.

**OPTIONS Enumeration**

The OPTIONS Request is supported by all user agents and servers present in a VoIP network, as dictated by [49], and this makes OPTIONS Scanning the most effective - and stealthy - enumeration method. An OPTIONS Request should either return a 200 (OK) or a 404 (Not Found) SIP Response. In most cases an invalid username in the OPTIONS Request will return a 404 response, informing the hacker that the username does not exist or the user is not logged in, while a valid username will return a 200. By trying a list of all possible usernames the
hacker will be able to enumerate the usernames of the target VoIP network. Again, depending
on the vendor of the VoIP gear probed, the response might be a 200 in both cases, so, in such
a situation, the OPTIONS method will not be effective.

As a proof of the varying effectiveness these methods of enumeration have it should be
mentioned that, as demonstrated in [36], REGISTER Enumeration is effective against an Aster-
isk Server but not against a SIP ExPress Router, while OPTIONS Enumeration and INVITE
Enumeration do work against a SIP ExPress Router but cannot yield any information when
probing an Asterisk Server.

SIPSCAN is a utility that can be found at [35] and which automates the SIP enumeration
process, combining all three methods described above. If targeted against a SIP proxy it will
return the valid usernames/extensions that it finds. If targeted against a SIP phone of which
we only know its IP address, SIPSCAN will return the extension that the phone uses to login
to the SIP Proxy. Knowing the exact extension assigned to a specific phone is vital for some
advanced attacks.

Preventing the enumeration of a SIP deployment is only possible to an extent. Segmentation
of the VoIP network on VLANs separate from data networks is an effective way of mitigating
many threats, including enumeration. Additionally, the SIP Proxy should be configured to
authenticate users before allowing the use of registration and other services. Finally, VoIP
Intrusion Prevention devices (available from various vendors like TrippingPoint, Sipera and
BorderWare) should be deployed at various locations in the VoIP network. These devices can
detect unusually high rates of Requests to the SIP Proxy and block the offending host.

3.3.3 VoIP Support Services Enumeration
Other than enumerating the VoIP user agents and servers, an attacker targeting the VoIP
network of an organization will also try to enumerate some core support services of the VoIP
infrastructure. Much harm can be done if those services are exploited.

TFTP Servers Enumeration

Trivial File Transfer Protocol (TFTP) servers are essential parts of most VoIP installations.
Many phones need to connect to the TFTP Server on boot-up to get their configuration files,
thus the server has to be exposed to the network. Taking into account the fact that the TFTP
protocol requires no authentication to download or upload a file and by recalling how easy it
was to Google Hack the address of TFTP servers during the Footprinting stage, one can easily
see why these servers are often the weakest link of a VoIP network security-wise.

The first thing a hacker would do is run Nmap, scanning for devices that listen on UDP
port 69, which is the TFTP port. This can be used to identify the IP address of the TFTP
server - if it was not found earlier - but it will also give her a list of all the devices that listen
on the TFTP port along with their MAC addresses. These addresses will come in handy
when trying to enumerate the configuration files available on the server. Since TFTP does
not support directory listings, the only way of getting the files the server holds is by knowing
their exact name. What makes MAC addresses valuable is that, sometimes, the name of the
configuration file the phone tries to download on power-up is a derivative of the phone’s MAC
address. Consequently, the hacker can combine the MAC addresses she found with the know
configuration filenames that common VoIP devices use (and which can be found on online lists
like the one at [35]) and, thus, through trial-and-error she will be able to download many of the configuration files present on the TFTP server. In each of these configuration files she will find everything there is to know about the specific device, including the SIP username and password.

Removing TFTP servers from the VoIP network would be the most effective countermeasure against the attacks described here, but, at least for the time being, many VoIP phones rely on these servers to operate. Newer phone models are starting to move away from TFTP and towards web-based configuration but TFTP cannot be abandoned just yet.

One possible option would be to use a firewall and restrict communications with the TFTP server, only allowing certain IP addresses to contact it. A determined hacker will be able to bypass the firewall though, since the source IP address of UDP packets can be spoofed. Moving the VoIP infrastructure on a separate switched VLAN can help mitigate the dangers of TFTP enumeration - and many other threats for that matter.

**SNMP Enumeration**

The inherent security problems of SNMP, used in many VoIP devices, have been discussed previously in this chapter. Using any SNMP probing tool of the ones mentioned earlier, like SNMPsweep, a hacker will be able to see which devices support SNMP and also if they responded to the default community strings, such as "public". If any phones use the default community strings, which is often the case, the hacker will proceed by using snmpwalk and the community string identified, in order to enumerate the configuration settings on the target phones. Once important information like the phone’s vendor is obtained, a vendor-specific SNMP MIB-targeted query can be launched, which can reveal even more information about the phones including the SIP username, SIP domain and DNS server address.

If possible, SNMP should be disabled on IP phones and for any other VoIP devices that use SNMP, version 3 of the protocol should be utilized. If using SNMP version 3 of is not an option, for backward compatibility or any other reason, then devices that use versions 1 and 2 should have the default public and private community strings changed.

**3.3.4 Enumerating VxWorks VoIP Devices**

VxWorks is an embedded real-time operating system that some IP phones are developed on. Among other things, it features a remote debugger which listens on UDP or TCP port 17185 and which, upon connection, allows for administrative debugging access to the device. Normally, this feature should be disabled before the phone actually ships but, unfortunately, many some vendors overlook that. ([92], [22], [33])

In this case, all an attacker has to do is use Nmap to scan the network for hosts listening on port 17185 and then use the native VxWorks debugger to connect to those hosts, thus gaining full administrative access to the devices.

Unless the vendor issues a patch or firmware update that disables the remote debugger, there is nothing the end-user can do about this threat. This is an example of an immature VoIP-oriented OS. Some compare the situation to the early years of data-network-based operating systems ([67]), but a link between the two cases is probably disputable since the Security and generally the whole IT sector has reached a considerably mature state nowadays.
Chapter 4

VoIP Network Threats

4.1 Eavesdropping

Privacy has always been a major concern for people. A brief study of the history of cryptography is enough to validate that. In the corporate environment, as already mentioned, voice can carry extremely sensitive information and a breach in confidentiality of voice communications is certainly a major issue. Offering a level of confidentiality that is, at least, equivalent to that of PSTN Telephony is one of the essential requirements for any corporate-wide VoIP deployment. This can be a hard task to accomplish since, as it has already been mentioned, it is generally considered easier to sniff traffic off a network than to tap a traditional phone line. Moreover, the fact that people using a VoIP phone see and use a device that resembles a regular phone makes them take for granted that equivalent privacy is indeed offered - even though it may not be the case - and therefore it is unlikely users will be cautious of what they say.

Since IP phone software runs on the application layer there are plenty of ways to secure VoIP traffic along the various network layers. Unfortunately, for the same reasons, there also various ways to sniff that traffic but for that to happen, access to the flow of VoIP traffic is necessary.

4.1.1 Gaining Access to VoIP traffic

Just like the information gathering techniques described in Chapter 3, the techniques a hacker might use to gain access to the target network are subject of general network security research and extensive information can be found in the relevant bibliography. Recommendations can be found in the prologue of the third chapter.

The techniques that a hacker might use to gain access to VoIP traffic include, but are not limited to:

- **Cable Tapping.**
  If an attacker can get physical access to cabling she will be able to install a number of sniffing devices, depending on the type of cabling. [126]

- **Sniffing Wi-Fi traffic.**
  It is trivial to capture WiFi traffic. Depending on the configuration, traffic might not be encrypted or the encryption could be weak. Additionally, it is hard to restrict the signal to building perimeter.

- **Connecting to a Hub**
  Hubs broadcast traffic traversing the hub to all ports. If an attacker manages to connect
or can compromise a PC connected to a hub, she can sniff the traffic of all the devices connected on the same hub.

- **MAC Flooding a Switch**
  Switches have a limit in the number of ARP/MAC index entries they can store. When that index fills, some switches revert to hub operation, thus transmitting data to every port. An attacker can send a large number of spoofed gratuitous ARPs to the switch, hoping it will revert to hub mode when flooded, which would make sniffing trivial. There is a number of automated tools for MAC address flooding, like Angst\(^1\).

- **ARP Poisoning**
  An attacker sends a number of gratuitous ARPs claiming that the Network Interface Cards (NICs) that have the IP addresses of target hosts assigned to them all have her MAC address. Keeping a relay index of the true MAC/IP address association she will be able to perform a *Man-in-the-Middle* (MITM) attack, during which all traffic between affected hosts will be routed to their destination but only after passing through her computer. This attack and some countermeasures will be discussed in more detail later.

- **Compromising a Network Node**
  If, by exploiting vulnerabilities in applications, or even in the operating system, running on a network node (phone, proxy, gateway etc.), a hacker manages to compromise the target node, then there are plenty of ways to capture live traffic. For example, a hacker might be able to use the administrative features of a compromised phone to capture all conversations terminating or originating from that phone. In case an attacker compromises a gateway, she will be able to remotely reconfigure that gateway to monitor all traffic that goes through it.

There several sniffing tools that can be used to capture wired or wireless traffic and one of the most popular utilities for the task is *Wireshark*\(^2\). An ARP Poisoning MITM attack can be automated using the *dsniff*\(^3\) tools.

Countermeasures against the aforementioned network sniffing techniques, like ensuring that access to wiring and network device closets is restricted to authorized personnel only and regularly scanning the network for devices that operate in promiscuous mode, can be found in various network security related resources.

Depending on the position of the compromised host, an attacker will be able to sniff different types of traffic and the type of eavesdropping she will be able to perform will vary accordingly.

### 4.1.2 Number Harvesting & Call Tracking

Passive number harvesting is relatively easy in a SIP environment. The simplest way is by sniffing all UDP and TCP traffic on port 5060 and looking for the From: and To: fields in the header. A more efficient way is to use the *Wireshark* sniffing utility which comes with a feature, activated by selecting Statistics -> VoIP Calls in the program menu, that automatically extracts all the VoIP call information, including phone numbers and URIs, from captured traffic.

\(^1\)URL: http://freshmeat.net/projects/angst/
\(^2\)URL: http://www.wireshark.org/
\(^3\)URL: http://monkey.org/~dugsong/dsniff/
Much like number harvesting, call tracking can be performed by simply examining all UDP and TCP port 5060 traffic. There are tools than can automate this process, though. VoIPon⁴ features, among other functions, the ability to log all calls to and from various IP addresses. Calling pattern information can also be acquired using Wireshark, as described above.

Number harvesting and call-pattern tracking can be dealt with by enabling encryption of signaling traffic. This can be done either on the network layer, using IPSec, or on the transport layer, using SIP TLS.

### 4.1.3 TFTP Configuration File Sniffing

The use, importance and vulnerabilities of TFTP servers have already been discussed in the Enumeration section of the third chapter. An attacker can launch Wireshark, set it to sniff all traffic in UDP port 69 and then search for the names of configuration files in the captured traffic. After this is done, there is nothing to stop her from downloading these files from the TFTP server and inspecting them for useful information, like usernames and passwords.

A separate VLAN, dedicated to the communication of the phones with the TFTP server could help mitigate the configuration file sniffing threat. Configuring a firewall to limit access to the TFTP server allowing only specific IP addresses belonging to VoIP Phones to connect, can also help.

### 4.1.4 Conversation Eavesdropping and Analysis

The number of tools that offer automated VoIP call eavesdropping features is far from negligible. Furthermore, their ease of use is astonishing. Wireshark can extract and analyze RTP streams from captured traffic and save them in various audio formats. Cain & Abel⁵, a popular sniffing and password cracking tool, also comes with a number of VoIP hacking features. VoIPon, mentioned earlier, can also extract conversations and save them in .wav files. Other than general-use network sniffing utilities, there are also tools designed specifically for H.323 VoIP deployments, like Fireberd DNA-323⁶ which, among its features includes the option to capture IP telephony conversations, as demonstrated in [80].

The only way to ensure the Confidentiality of VoIP conversations is to encrypt the RTP media stream. This can be done on the network layer, using IPSec, in the same way that signaling traffic is encrypted, or on the transport layer using Secure Real-Time Transport Protocol (SRTP, defined in RFC 3711 [52]). A newer media encryption protocol is ZRTP([39]) which, in essence, is an enhancement on SRTP. Quoting the draft submitted in IETF on July 2007, ZRTP is "a protocol for media path Diffie-Hellman exchange to agree on a session key and parameters for establishing Secure Real-time Transport Protocol (SRTP) sessions". So far, though, its only implementation is the softphone encryption plugin Zfone⁷. It is essential to point out that, as discussed in the first chapter, in some cases regulators may find that the use of ZRTP does not comply with various lawful interception laws that are in place in various parts of the world. These laws include UK’s Regulation of Investigatory Powers Act 2000 (RIPA)⁸ and

---

⁴URL: http://www.enderunix.org/voipong/
⁵URL: http://www.oxid.it/cain.html
⁷URL: http://zfoneproject.com
the Communications Assistance for Law Enforcement Act (CALEA)\(^9\), a United States lawful interception law which was passed in 1994 (Pub. L. No. 103-414, 108 Stat. 4279) with the purpose: "To amend title 18, United States Code, to make clear a telecommunications carrier’s duty to cooperate in the interception of communications for Law Enforcement purposes, and for other purposes". Consequently, in some parts of the world, ZRTP might not be a real option for corporate deployments where interconnection with the PSTN service is required. Regulatory compliance is an important factor to consider for all organizations.

It should also be noted that it is not necessarily appropriate to apply end-to-end encryption on both the signaling and media streams since it adds overhead that can affect the QoS (depending on the quality of the underlying network). Encryption over untrusted parts of the network and over the Internet should be sufficient in most cases.

4.1.5 DTMF Decoding

A less obvious danger that arises from having a hacker listen in on VoIP communications is DTMF decoding. Some voice services, such as Phone Banking, require that the users type in sensitive information like PIN or account numbers. An attacker who captures that call can then replay it and, using the *DTMF Decoder*\(^{10}\) utility, convert the recorded tones to the corresponding digits, thus getting hold of the aforementioned sensitive information.

Although there are no easy-to-implement safeguards available, extending the ZRTP protocol to protect DTMF or sending DTMF through protected signaling are some potential countermeasures. [113]

---

\(^9\)URL: http://www.fcc.gov/calea/

\(^{10}\)URL: http://www.polar-electric.com/DTMF/Index.html
4.2 Denial of Service Attacks

Before describing Denial of Service (DoS) attacks, or in other words attacks that can affect the availability of IP telephony services, it is essential to examine exactly how sensitive this technology is to network performance variations.

4.2.1 VoIP Quality of Service

The criticality of the Quality of Service (QoS) in a VoIP deployment has been mentioned briefly in the introductory chapters but a more detailed look will help understand how easy it can be to perform a DoS attack on a VoIP network. Network availability should, of course, be guaranteed before even thinking of moving to the deployment of VoIP. Network availability, network security and QoS are difficult enough to guarantee as it is and this task gets even harder in the case of VoIP. The time-critical VoIP applications are very sensitive to network bandwidth issues and certainly way more sensitive than traditional data applications. This can also affect the arsenal of security mechanisms that network administrators have in their hands. Most traditional firewalls and encryption devices cannot be used in a VoIP network. Such devices simply add to network congestion and throughput delay in a prohibitive to the operation of VoIP way.

A list of the main factors that affect VoIP call quality follows:

- **Network Latency**
  Latency refers to the amount of time it takes for a VoIP transmission to travel from its source to its destination. Latency can be affected by each intermediate host between the two endpoints, length of cabling that has to be traversed, network traffic, time it takes to encode and decode voice etc. Both the International Telecommunication Union (ITU) and NIST recommend a maximum one-way latency of 150ms (in [66] and [75] respectively). The fact that the IP telephony service is extremely sensitive to latency variations makes the technology extremely vulnerable to Denial of Service attacks and even more vulnerable to Degradation of Service attacks ([99]). More on that in later chapters.

- **Jitter**
  Jitter occurs when the listener receives voice packets with variable delays. This is not uncommon in the packet-switched where data packets sometimes reach the destination through different paths. Jitter can cause packets to arrive in mixed order and be processed out of sequence. During the call, any jitter higher than 25ms will be noticeable ([36]). To reduce the effects of jitter many VoIP applications and devices utilize a jitter buffer where a number of packets is stored before processing. Finding the optimal size for that buffer is crucial, otherwise it might cause a rise in the communication latency, as demonstrated in Figure 4.1, taken from [62].

- **Packet Loss**
  Under heavy load networks might randomly drop data packets and retransmit them. This is not acceptable in VoIP networks since in this case retransmitting the data is not an option. By the time a voice packet is retransmitted, the conversation will have moved on and that data is will be useless. Taking into consideration that a single packet might contain 40 to 80ms of voice, depending on the compression method used, it is not hard to understand why even a packet loss of 1% can severely degrade the QoS of the VoIP service [120]. Although the effect of packet loss is significant in all cases, it should be noted that it varies considerably depending on the codec used, as demonstrated in [103].
The Telephony Sector of the International Telecommunications Union (ITU-T), in its G.113 ([65]) and G.114 ([66]) Recommendations states that in order to provide similar call quality to PSTN, a VoIP network must exhibit the following metrics: latency less than 150ms, jitter less than 25ms and less than 5% packet loss. It must be pointed out that jitter is the factor that affects listening quality most. Latency can cause large delays but users, although frustrated at first, manage to adapt after a while and can still communicate - even if it is in a "single duplex" mode where only one party speaks at a time. Therefore, in a congested network the use of a large jitter buffer, at the cost of increased delay, can improve listening quality ([57]).

Figure 4.2, taken from [36], shows the maximum latency, jitter and packet loss various ISPs guarantee to their customers, based on the values listed in their Service Level Agreements (SLAs).

![Figure 4.1: Jitter Increases Latency](image-url)
There are many solutions that help guarantee a certain level of QoS for VoIP applications. One of the most commonly used approaches to QoS is Differentiated Services (DiffServ), which involves tagging all packets with a priority label, called Differentiated Services Code Point (DSCP), which is applied at the IP layer and is based on the type of application the packets originate from. This allows network devices to prioritize incoming packets accordingly. For example, RTP packets would have higher priority compared to email traffic. More details on the DiffServ QoS solution can be sought out in [19].

To summarize, the challenge in ensuring VoIP QoS lies in having each communication endpoint receive the transmission media in a predictable and quantifiable way, despite the fact that this media is connectionless in nature and, moreover, has to compete with data traffic present on the network ([117]). Further information and recommendations on the issue of VoIP Quality of Service can be found in [65], [66], [62] and [75].

### 4.2.2 Flooding Attacks

It should be clear by now that making VoIP calls sound as clear and generally offer the same QoS as PSTN calls do, while maintaining the confidentiality, integrity and availability of the network, is a tough challenge since even a small deterioration in network performance can affect user experience. During a DoS Flood Attack an attacker launches a flood of packets that consume network and/or system resources so that they are not available to legitimate users or available with significantly degraded quality. If the flooding attack is launched from a single machine it is relatively easy to isolate the offending machine and drop all packets coming from

<table>
<thead>
<tr>
<th>ISP</th>
<th>Latency</th>
<th>Jitter</th>
<th>Packet Loss</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT&amp;T Managed Internet Service</td>
<td>39 ms</td>
<td>Not given</td>
<td>0.1 percent</td>
</tr>
<tr>
<td>(<a href="http://www.att.com/abs/serviceguide/mis/sg_mis_service_lvl_mgmt.html">http://www.att.com/abs/serviceguide/mis/sg_mis_service_lvl_mgmt.html</a>)</td>
<td>maximum latency</td>
<td></td>
<td>maximum packet loss</td>
</tr>
<tr>
<td>Verizon Voice over IP</td>
<td>55 ms</td>
<td>1 ms maximum</td>
<td>0.5 percent</td>
</tr>
<tr>
<td>(<a href="http://www.verizonbusiness.com/terms/us/products/advantage/">http://www.verizonbusiness.com/terms/us/products/advantage/</a>)</td>
<td>maximum latency</td>
<td></td>
<td>maximum packet loss</td>
</tr>
<tr>
<td>Qwest SLA</td>
<td>50 ms</td>
<td>2 ms maximum</td>
<td>0.5 percent</td>
</tr>
<tr>
<td>(<a href="http://www.qwest.com/legal/docs/Qwest_Internet_SLA_10_07_04_.pdf">http://www.qwest.com/legal/docs/Qwest_Internet_SLA_10_07_04_.pdf</a>)</td>
<td>maximum latency</td>
<td></td>
<td>maximum packet loss</td>
</tr>
<tr>
<td>Verio SLA</td>
<td>50 ms</td>
<td>Maximum jitter</td>
<td>0.1 percent</td>
</tr>
<tr>
<td>(<a href="http://www.verio.com/global-ip-guarantee/">http://www.verio.com/global-ip-guarantee/</a>)</td>
<td>maximum latency</td>
<td></td>
<td>maximum packet loss</td>
</tr>
<tr>
<td>Internap SLA</td>
<td>45 ms</td>
<td>Less than 0.5 ms</td>
<td>0.3 percent</td>
</tr>
<tr>
<td>(<a href="http://www.internap.com/product/technology/performanceip/page1525.html">http://www.internap.com/product/technology/performanceip/page1525.html</a>)</td>
<td>maximum latency</td>
<td></td>
<td>maximum packet loss</td>
</tr>
</tbody>
</table>

Figure 4.2: Maximum Latency, Jitter and Packet Loss defined in Service Level Agreements
the specific IP address, thus effectively mitigating the attack. This is one of the main reasons that have led to the emergence of Distributed Denial of Service (DDoS) attacks which involve many attacking machines, usually a whole botnet of compromised computers, which the attacker controls to attack the target network. Several types of Flooding attacks that can be launched against the VoIP network will be examined. Plenty of tools to automate these attacks can be found on the DoS tools sections of the Packet Storm security website.\footnote{URL: http://www.packetstormsecurity.org/DoS/}

**TCP SYN Flood**

A TCP SYN Flood attack involves exploiting the mechanics of a TCP connection handshake to overwhelm the target with SYN requests. The attacker creates SYN flagged packets, each with a spoofed, random, IP address and sends the to the target host. The victim responds to every SYN request with a SYN-ACK packet and then waits for some time to get an ACK packet as a response. Since the source IP address is spoofed that ACK packet is never received and soon enough the target machine fills up its connection table with bogus SYN requests, unable to respond to legitimate SYN packets. \cite{14} provides more information for this type of DoS attack.

**UDP Flood**

*User Datagram Protocol* (UDP) source address can easily be spoofed and, moreover, it is supported by most SIP-based VoIP devices. This makes UDP flooding an attractive DoS method for attackers. A UDP Flood attack takes place when an attacker sends UDP packets with spoofed source address to random ports of the target host. The victim has to check for applications listening on that port and, when it figures out there is nothing running on that specific port, respond with an ICMP packet to inform the sender. Because of that, if a flood of UDP packets are sent to the target VoIP device, it will probably go down. There are various tools that launch UDP Flood attacks, such as *UDPFlood*\footnote{URL: http://www.foundstone.com/us/resources/proddesc/udpflood.htm} stress-testing tool from *Foundstone*.

**Ping Flood**

This is a simple attack that involves sending numerous *ICMP Echo* (Ping) requests to a target host, hoping to overwhelm it with traffic.

**Smurf Attack**

*Smurf* attacks can be considered the evolution of regular Ping floods. Such an attack involves spoofing *ICMP Echo* requests to present the victim’s address as the source of these request and then sending them to numerous random hosts. Those hosts will then legitimately answer to these requests, flooding the victim with their responses. A detailed description of Smurf DoS attacks can be found in \cite{13}.

**Worms**

Taking into account how sensitive IP telephony is to network health degradation, even a worm outbreak can be the cause of a Degradation of Service or even a Denial of Service attack. Worms scanning for vulnerable hosts and propagating through the network certainly affect network health and, consequently, VoIP QoS. Depending on the type of worm and the number of vulnerable hosts, an attacker could succeed in launching a Denial of Service attack simply by injecting a worm into the network.

**QoS Manipulation**
In case an organization’s VoIP network features QoS mechanisms such as DiffServ mentioned earlier, most of the internal flooding attacks would perhaps manage to degrade VoIP service quality but, since RTP traffic would be prioritized, a full scale DoS attack would be hard to realize. A more sophisticated attack would involve diverting the QoS mechanisms, simply by flooding the network with high priority traffic; RTP traffic. If an attacker can flood a phone, PBX or proxy with RTP traffic, the QoS would be unable to help since they would have no means of discerning legitimate from bogus traffic. This might go as far as actually calling a specific IP phone which is the target of the DoS attack, but more on this in the Session & Application Threats chapter that follows.

There are plenty of ways to mitigate DoS and DDoS attacks, including security appliances designed for this purpose. Vendors like TippingPoint\(^\text{13}\) offer appliances that can be deployed in the network and detect and mitigate DoS attacks, distributed or not. Additionally, a network administrator can ensure her network perimeter is hardened and configured to resist DoS and DDoS attacks enforcing specific network policies such as SYN rate limiting and ICMP blocking. Moreover, IP phones and the various servers present in the VoIP deployment should be hardened. As already mentioned, changing default passwords, disabling unnecessary services and keeping every device and application, including operating systems, patched is essential. The importance of having core VoIP servers and devices on a separate VLAN is again evident, when trying to protect from the worms and other threats that might plague the regular, less sensitive, data network.

4.2.3 Network Availability Attacks

Instead of just trying to flood the target with traffic, an attacker can try to crash the target network device and/or its operating system using more sophisticated attacks.

Fuzzing

VoIP device vendors implement the network stack in various ways. Not all implementations feature the same robustness. Often developers do not consider the possibility that network input might deviate from what is expected and this may cause devices and applications to crash upon processing input that differs from the standard expected. Crafting various different packets that deviate from what the target protocol defines and trying to find bugs and vulnerabilities in the protocol implementation that will cause it to malfunction or crash, is called fuzzing. There are various tools that can be used for such attacks (or “stress tests” - depending on who uses the tool and for what purpose), called Fuzzers, such as the IP Stack Integrity Checker (ISIC)\(^\text{14}\) which can used to check the stability of the IP, TCP, UDP, ICMP and Ethernet stacks. Fuzzing will be discussed in more detail in the next chapter.

Packet Fragmentation

Sending fragmented TCP and UDP packets can also affect the operation of systems and VoIP devices. There are quite a few variations of this attack and there are automated tools that can be used to launch such attacks. An example of such a tools is Teardrop\(^\text{15}\), which exploits the fact that some implementations of the TCP/IP IP fragmentation re-assembly code cannot deal with overlapping IP fragments (see [10]).

\(^\text{13}\)URL: http://www.tippingpoint.com/
\(^\text{14}\)URL: http://www.packetfactory.net/projects/ISIC/
\(^\text{15}\)URL: http://packetstormsecurity.org/Exploit_Code_Archive/teardrop.c
Operating System & Firmware Vulnerabilities

Other than attacking the various protocol implementations used in VoIP devices and applications, an attacker might try to aim even lower and go for the underlying operating system or firmware of the target. If, for instance, an Asterisk application is run on a Linux server and there is a vulnerability that affects the specific Linux kernel version that the server runs, an attacker can target the operating system, exploit the vulnerability and cause a VoIP DoS without having to deal with the VoIP application itself. This is a serious threat to the availability of a VoIP deployment which might include an assortment of IP phone models, perhaps even coming from different vendors, and a number of servers, with many of these devices utilizing different firmwares and operating systems. A good example is the vulnerability in the Distributed Component Object Model (DCOM) Remote Procedure Call (RPC) interface of the Windows operating system which allowed the W32.Blaster Worm to attack the Cisco CallManager server and other Cisco VoIP applications (see [20]).

In addition to the countermeasures already described in the Flooding Attacks section, Intrusion Prevention/Detection Systems (IPS/IDS) can significantly help to deal with network availability attacks. These network devices can detect potentially malicious traffic such as fragmented packets or traffic patterns that indicate an attack and block the source, while allowing legitimate traffic to go through. A detailed look into IDs and IPS systems is beyond the scope of this Thesis. [105] is an extensive guide to the subject.

4.2.4 Supporting Services Attacks

Besides various VoIP devices, servers and PBXs, a corporate-wide VoIP deployment will have to rely on some infrastructure network services such as DHCP and DNS. If one of these services is brought down, VoIP applications might demonstrate limited usability.

DHCP Exhaustion

The Dynamic Host Configuration Protocol (DHCP), defined in [31], provides automatic allocation of IP addresses and additional configuration options to devices that connect to a network. Many VoIP phones support DHCP and, by default, request an IP address on bootup. This means that, depending on the phone configuration, if, for any reason, the DHCP server is unavailable or has already assigned all available IP addresses to other devices, the phone might not be usable. DHCPX, which is included in the BackTrack2 penetration testing linux distribution, is a DHCP Flooder. In other words, an attacker can use this tool to exhaust all the IP addresses available to the DHCP server, so that no new devices that rely on DHCP can get an IP address.

The threat of DHCP Exhaustion attacks can be mitigated, to an extent, if the DHCP server is configured in such a way that only hosts with known MAC addresses and originating from trusted network segments can be given an IP address. Additionally, VoIP DHCP servers should be kept on a different VLAN to that of traditional data traffic ([72]).

DNS Cache Poisoning

Some details about the way DNS works have been discussed in Chapter 3. A DNS Cache Poisoning attack entails tricking a DNS server into believing it has received an authentic DNS response, while that response is actually fake. This could mean that all users served by the victimized DNS server will be redirected to an address different to the one they were actually looking for. DNS Cache poisoning can pose a serious threat to large-scale VoIP deployments.

\[16\] URL: http://www.remote-exploit.org/backtrack.html
that cover different network domains. In such cases DNS SRV records are used during SIP phone dialing to locate the domain of the destination SIP Proxy (see Figure 2.2). If an attacker manages to alter these database entries then she would, for example, be able to redirect all calls coming to the target organization’s domain to a SIP proxy of her choice. This is certainly not a pleasant situation for any organization and can enable many other threats, such as VoIP Phishing (Vishing) which will be discussed later on. DNSA\textsuperscript{17} is a DNS Auditing tool that can assist the attacker in performing this type of attack.

DNS Cache Poisoning should not be a threat if the DNS servers are configured properly. Detailed information and recommendations on how to secure a DNS server can be found in [58].

**DNS Flood DoS**

All the Flooding DoS techniques mentioned in the previous section can be launched against a DNS server, thus consuming all bandwidth and/or available connections and depriving the VoIP network of the DNS service. UDP Flood attacks in specific can pose a significant threat since many firewalls cannot discern malicious traffic from legitimate DNS requests and responses.

**Operating System & Firmware Vulnerabilities**

Just like any other network device, servers that run supporting services are open to a number of attacks which exploit vulnerabilities present in the underlying Operating System and/or Firmware. Authentication, TFTP, DNS and DHCP servers could all be targets of such attacks.

It should be evident at this point that besides securing and keeping patched and up-to-date all of the VoIP devices and servers, it is also essential to protect the entire network infrastructure on top of which VoIP is running.

\textsuperscript{17}URL: \url{http://www.packetfactory.net/projects/dnsa/}
4.3 Interception & Modification Attacks

If an attacker manages to intercept and modify VoIP signaling or voice traffic, there are plenty of harmful attacks she can launch. The mere fact that a single packet can contain up to 80ms of recorded and compressed voice can give a good estimate of the deterioration in call quality that an attacker can cause simply by reordering or dropping a few packets.

4.3.1 Man-In-The-Middle

A Man-In-The-Middle (MITM) attack is one in which the attacker (who might be a legitimate network user or an external attacker who managed to secure access in some way) succeeds in becoming an intermediate node between two communicating hosts, without the hosts realizing this is the case. In the case of a VoIP network the victims could be two phones or a phone and a SIP proxy and so on. The traffic generated by conversations would then be routed through the attacker’s computer and she would be able to eavesdrop on the conversation, as already mentioned in the Eavesdropping section. Moreover, the attacker could decide to move on to a more active role and drop some packets which would certainly affect the QoS, maybe replay or insert media, thus altering the conversation, and even dropping all packets, causing a DoS. In addition to the aforementioned threats, and depending on the network location in which the hacker managed to insert herself, more attacks can be realized, such as DNS Spoofing and DHCP Spoofing.

ARP Poisoning

ARP Poisoning, which was mentioned earlier as one of the ways to eavesdrop on VoIP calls, is probably the most common method used to perform a MITM attack. This method is possible because, unfortunately, many operating systems will replace or accept ARP entries even if there was no ARP request sent. Additionally, no authentication is used at any stage of the IP address-MAC address mapping.

The basic mechanics behind ARP Poisoning have already been discussed so an example would be more appropriate at this stage. Assuming the attacker wants to insert herself between an IP phone and a SIP proxy, she would first have to find out the MAC addresses of the two targets. Then she would send a gratuitous ARP to the proxy claiming that the MAC address for the target phone has changed and the new MAC address is the one of her computer. An equivalent gratuitous ARP would be sent to the phone, claiming that the proxy’s MAC has changed to her MAC. As a last step, she would enable IP forwarding on her computer, routing traffic to its intended destination so that the victims do not realize there is a MITM attack in progress.

ARP poisoning functionality is included in many hacking tool collections and penetration testing suites. Some popular utilities that could be used for the task are: Cain & Abel\textsuperscript{18}, which features a number of other VoIP attacks as well, ARPspoo\textsuperscript{19} which is included in the dsniff\textsuperscript{19} tool collection and ettercap\textsuperscript{20}, a MITM/Sniffing Suite.

The significance of using a VLAN that will logically separate the VoIP deployment from the standard data network has to be pointed out once more. Moreover, using static ARP entries, which means manually entering the MAC-IP mappings for every host on the network, is one solution that could help mitigate the ARP Spoofing threat. This is usually done at the switch

\textsuperscript{18}URL: http://www.oxid.it/cain.html
\textsuperscript{19}URL: http://monkey.org/~dugsong/dsniff/
\textsuperscript{20}URL: http://ettercap.sourceforge.net/
but for some critical servers, like proxys and gateways, it might be a good idea to set static mappings on the underlying operating system of those network devices as well. As an extra layer of security, there should be specific MAC addresses that are allowed to connect to switch ports. Both of these measures can add too much management overhead in larger networks, but, nevertheless do help in securing the network against ARP poisoning attacks. It should be reminded that various VoIP encryption methods are available that can protect signaling and voice and can be applied to various network layers. There is SRTP and ZRTP on the application layer, TLS on the transport layer as well as IPSec on the network layer, all of which can protect a VoIP deployment against this and other types of attacks. Finally, the arpwatch utility can prove a valuable tool in the hands of a network administrator since it can detect ARP poisoning attacks. It keeps track of IP/MAC mappings and can report any suspicious activity via logs or email.

It should be pointed out that there are various academic suggestions on the subject of securing ARP. One such suggestion is the Secure Address Resolution Protocol (S-ARP), presented in [9], which proposes an extension of ARP which will utilize an authentication scheme for ARP replies using public key cryptography, thus rendering ARP Poisoning attacks ineffective. Another solution suggests securing ARP via the combined use of digital signatures and one time passwords based on hash chains ([48]).

4.3.2 Application Layer Interception

Moving on to the application layer, there are various techniques an attacker can use to realize the same threats to the confidentiality, integrity and availability of a VoIP network as the ones described in the previous section. Not only that but, working at the application layer, even more dangerous attacks can be carried out, since the sophistication of the tools used is higher. Operating on the application layer the malicious software can see and relay both signaling and media traffic, thus can do almost anything to this traffic.

Application-level interception means using rogue applications that can trick the target and make it think it communicates with a legitimate VoIP device. This requires both finding a way to trick the target SIP phone or SIP proxy into communicating with a rogue application and also having a rogue application that can successfully imitate the behavior of the legitimate SIP phone or SIP proxy.

Inserting Rogue Application

An overview of possible ways to insert the rogue application into the target SIP network, in a way that it looks like its a legitimate device, follows:

- Physical Access
  If an attacker has physical access to the network things are straightforward as she can easily connect her own PC to the network.

- Man-In-The-Middle Attack
  The MITM attacks described previously can be used to trick a SIP phone or proxy into talking with a rogue application.

- SIP Phone Reconfiguration
  If the attacker manages to gain physical or remote access to a SIP phone configuration, she can change the SIP proxy address to the address of the rogue proxy. This way the next time someone tries to call using the specific device it will be routed through the rogue proxy and not the legitimate one.

21URL: http://www.securityfocus.com/tools/142
Redirection Response Attack
There are certain responses to a SIP INVITE that can cause inbound calls to be redirected to a rogue phone. More on this in the VoIP Session & Application threats chapter that follows.

Registration Hijacking
Replacing the registration of a legitimate phone to the SIP proxy with a registration that includes the address of the rogue application instead, all inbound calls will be redirected to that rogue application. Again, more on this technique in the VoIP Session & Application threats chapter.

So, it is relatively easy to insert a rogue SIP application into the target network. What requires a higher level of sophistication is to actually find and configure this application.

Rogue SIP Application
As already stated, depending on the situation, an application that can effectively mimic a SIP phone or an application that can mimic a SIP proxy or an application that can do both, is required.

In the case of rogue SIP phone application, the attacker will be able to see and manipulate all signaling and media traffic that is intended for the legitimate phone for which the rogue application acts as a gateway - without the victim knowing about it of course - at her will. The situation is even more dangerous in the case of a rogue SIP proxy application. The worse that can happen in this case is to have the attacker manage to position her rogue proxy application between two legitimate proxies. This would mean the attacker would be able to see and manipulate all traffic between the two proxies which, in cases of a large-scale VoIP deployments, could translate to thousands of calls.

An application that, among other things, can act both as a SIP proxy and as SIP phone, depending on how it is configured, is sip_rogue\textsuperscript{22}. With this application in her hands an attacker could launch various types of extremely dangerous attacks, with the option to choose which side, if not both, will be affected from the attack. Some examples would include listening in on calls real-time and/or recording them, replacing conversation audio stream with an audio stream of her choice or even adding audio stream to the legitimate one (e.g. adding background noise to the conversation the victims have), dropping calls depending on the time of day, who is calling and so on. Of course all the of the threats enabled by eavesdropping, such as DTMF and calling pattern monitoring, are present here as well. When operating as a SIP Proxy the arsenal of attacks at the hacker’s disposal is even larger since in this case she would have the option to redirect calls, force the IP phone not to use media encryption, monitor for passwords and other interesting information etc. Some examples and more details on the use of sip_rogue can be found in [36] and in the documentation that comes with the application.

If an attacker manages to launch such an attack, the impact may vary. Depending on the position that the attacker deploys the rogue application it might affect one or many users. Some attacks, such as changing the contents of a conversation by replacing the audio stream, can have disastrous effects. Another important factor is the position of the victim in the target organisation. A hacker listening in and recording the calls of the Executive Officers is not something any organization would accept, regardless of its risk appetite.

\textsuperscript{22}URL: http://www.hackingvoip.com/tools/sip_rogue.tar.gz
For the above attacks to be feasible the attacker has to have access to the target organization’s internal. It has already been shown in the introductory chapters how an attacker can find Internet-accessible systems. Of course, in the case of large-scale VoIP deployments that have calls routed through the Internet, it is even harder to protect from public exposure of some parts of the network. Other ways into the network such as physical access should not be overlooked and the security policies should adapt and be implemented accordingly (see [25]). The best way a VoIP network administrator can deal with such attacks is to make sure that none of the techniques a hacker might use to insert a rogue SIP application are effective.
Chapter 5

VoIP Session & Application Threats

5.1 Fuzzing

Having all systems patched and up-to-date can only guarantee a certain level of safety but, of course, a network can never be considered 100% secure. If an attacker cannot find any known vulnerabilities in VoIP (and other) network devices to exploit, she might have to resort to black box testing the target. The term refers to security testing with no prior knowledge about the target’s internal structure (e.g., no source code available). One common approach to black box testing is Fuzzing, or Fuzz Testing, which was briefly mentioned when flooding attacks were discussed. This stress-testing and vulnerability-discovery technique was first developed in 1989 (for the original paper see [88]) and, in essence, it is about feeding the target application with random input and monitoring how it reacts, looking for any abnormal behavior (e.g., a crash) that might indicate the existence of a vulnerability in the protocol implementation or elsewhere.

Some types of vulnerabilities that may be discovered when an attacker or a security researcher runs fuzz tests include buffer overflows, integer overflows, endless loops and logic errors as well as various format string vulnerabilities. This is of course just a sample of the types of vulnerabilities that can be uncovered via fuzzing. The subject is touched in most network security books and, for a more in-depth look, there are whole books devoted to fuzzing, such as [112].

The tools that an attacker can use to launch fuzzing attacks, either to cause a crash and thus a DoS or to look for vulnerabilities that will allow her to compromise the vulnerable device, vary. The PROTOS group offers a SIP Fuzzing Suite, the PROTOS Test-Suite: c07-sip\(^1\) while detailed information on the subject of SIP fuzzing and instructions on how to use this tool can be found in the aforementioned URL. Ohrwurm\(^2\) is an RTP fuzzing tool which, because of the nature of the RTP protocol, requires a real-time conversation in order to work. In other words the attacker would have to perform a MITM attack before being able to stress-test the RTP implementation of the target VoIP device.

The PROTOS Project\(^3\) of Oulu University is probably the most successful university fuzzing research program. Their testing led to the discovery of vulnerabilities in many protocols including serious security holes in various implementations of the SIP (see [12]) and H.323 (see [11]) VoIP protocols. Another example of the effectiveness of fuzzing in VoIP security research is [94]. Naturally, as VoIP technology is becoming more widely deployed, research on its security

\(^1\)URL: http://www.ee.oulu.fi/research/ouspg/protos/testing/c07/sip/index.html
\(^2\)URL: http://mazzoo.de/blog/2006/08/25
\(^3\)URL: http://www.ee.oulu.fi/research/ouspg/protos/
issues is becoming a more attractive subject.

Arguably, the most effective weapon in the hands of VoIP administrators and security teams against Fuzzing attacks is fuzzing their own network before a malicious person does so. Of course, this has to be done in a non-production environment or under strictly controlled conditions. In case a vulnerability is found then the vendor should be informed immediately and extra security measures taken until a patch is available. The speed and effectiveness of the vendor’s response, as well as any Quality Assurance and Security Testing documentation available, will should affect the trust that is placed on the specific products. The deployment of an advanced Intrusion Prevention System (IPS) that can detect traffic not conforming to the protocol standards (e.g. format string flaws) and block it, can certainly help. What should be noted is that such a system might cause compatibility issues since many vendors decide to enhance protocol used and add more features in their implementation of it. There is a chance that the IPS will considered such traffic malicious activity and block it.
5.2 Denial of Service Attacks

Various flooding techniques that can be used to launch a Denial of Service Attack (DoS) have already been mentioned. These techniques were mostly based on network protocol manipulation. In this section, moving on to higher network layers, DoS flooding techniques that utilize various VoIP protocol and session messages will be covered.

These attacks can only be realized if the target device can be deceived into accepting accepting and processing malicious requests. If the target is flooded with such requests, then legitimate requests from other devices will be processed slowly, disrupting the targeted service, or even ignored, resulting in a DoS condition.

VoIP signaling can be run over UDP or TCP. UDP, although faster, is not as secure as TCP, mainly because of its connectionless nature. It does not provide any mechanism that would allow the identification of rogue traffic and its extremely vulnerable to flooding attacks. TCP on the other hand is a connection-oriented protocol which means that a persistent connection is setup between the communicating ends. This allows for the use of various mechanisms, such as sequence numbers, that make packet injection a harder task for the attackers. Of course, as mentioned in earlier chapters, it is not impossible, but it certainly is harder than in the case of the UDP protocol. Unfortunately, most vendors choose to use UDP on their VoIP devices.

5.2.1 SIP Proxy Attacks

The importance of the SIP Proxy in a VoIP network should be obvious by now. All requests between Media Gateways, SIP Phones and other VoIP resources are processed by the SIP Proxy. If it goes offline or is not able to process any requests, the entire IP telephony service could be disrupted. Several attacks that a hacker could launch against a SIP proxy, aiming to cause such a disruption or denial of service, are described next.

**SIP Proxy UDP (or TCP SYN) Flood**

The fact that most vendors choose to transport signaling over UDP leaves most SIP proxies open to UDP Flood attacks. By using a tool like udpflood\(^4\) an attacker can launch a udp flood on port 5060 - which will certainly be open on the target proxy since its the port the SIP service listens to. The impact of this attack can vary from a minor disruption of service, where a few legitimate users might experience trouble getting their calls through, to a full-scale denial of service where no calls can be made, depending on the SIP Proxy application vendor, as demonstrated in [36]. In the rare case that the SIP deployment uses TCP, the attacker could launch a TCP SYN Flood, the mechanisms of which were seen in the previous chapter.

**SIP Proxy INVITE Flood**

Perhaps it should be reminded that INVITE Requests are used to initiate calls. If a SIP proxy is flooded with INVITE requests the users of the IP telephony service might experience a disruption or denial of service since the proxy will not be able to cope with the processing of all the requests. Since these requests are typically sent over UDP, it is not hard for a malicious person to launch such an attack.

A useful tool that could be employed by the attacker is the linux-based inviteflood\(^5\), which sends a flood of INVITE requests to the proxy, each appearing as an independent call. For each INVITE the victimized proxy sends a response to the malicious device, the latter does not reply and that forces the proxy to retransmit its response - which makes things even worse. By

\(^4\)URL: http://www.foundstone.com/us/resources/proddesc/udpflood.htm
\(^5\)URL: http://www.hackingvoip.com/tools/inviteflood.tar.gz
manipulating the contents of the INVITE requests the attacker can stress the proxy server even further to guarantee some degree of degradation in the QoS even on more resilient servers. This also gives more options to the attacker. For example, in case the attacker does not know any legitimate SIP phone address she could launch the INVITE flood including a non-existent SIP phone address in the requests. It should be noted that these attacks are possible even in the rare case that the target SIP proxy requires authentication before it will forward any INVITE requests. In that case the Proxy will answer with a 407 Proxy Authentication response, but this will still add enough overhead to make it hard for the Proxy to cope with a flood of such requests. This also gives more options to the attacker. For example, in case the attacker does not know any legitimate SIP phone address she could launch the INVITE flood including a non-existent SIP phone address in the requests. It should be noted that these attacks are possible even in the rare case that the target SIP proxy requires authentication before it will forward any INVITE requests. In that case the Proxy will answer with a 407 Proxy Authentication response, but this will still add enough overhead to make it hard for the Proxy to cope with a flood of such requests.

Another variation of the Invite flood is to send requests that are supposed to be forwarded to another domain. If this domain is a random non-existent domain, the attack can be considered unsophisticated but it can still be very effective on some SIP proxy platforms which cannot detect that the destination proxy is non-existent and continue wasting resources attempting to establish calls that cannot be completed. This attack can become way more dangerous if the target domain is a real one. In this case, the request is forwarded from the first SIP proxy to the proxy of the target domain, which looks for the target SIP phone and responds accordingly. If the address of the SIP phone to be called is invalid, then it informs the original proxy with an error message, while if the address is valid the target IP phone rings. This can be a devastating attack (especially in the latter case where the phone keeps ringing) since it targets both Proxy servers and, additionally, the communication overhead and memory required to keep track of a flood of such INVITES is huge. Hence, it is extremely unlikely that the two Proxies will maintain their availability to legitimate users.

It should be noted that in all cases that a legitimate SIP phone address is used, the attacks affect the SIP phone as well and not just the Proxy (or Proxies) that handle the calls.

The successful launch of such an attack can have significant impact. A few missed calls might mean a lot if these missed calls are customers trying to get through to the support line or business partners calling to discuss about important new deals. A full-scale DoS where two or more SIP proxies go out of service and the organisation is left with no phone lines could be considered a disaster.

Separating the VoIP network from the underlying data network using VLANs and generally implementing the the DoS/DDoS countermeasures suggested in the previous chapter, are some obvious ways to protect - to some extent - a VoIP deployment. The superiority (security-wise) of TCP over UDP has already been pointed out. Most of the aforementioned attacks would be better mitigated in the case of a SIP deployment running on TCP since it would be harder to trick the target into accepting the packet flood. Mechanisms such as Sequence Numbers would guarantee that malicious injected traffic would be discarded at a lower protocol layer and never reach the SIP Proxy application. Of course, in order to have any of these benefits, TCP should be used in by all SIP devices. If a single phone among the whole network uses UDP, then that phone is the weakest link in the security chain. TCP also gives the options to employ Transport Layer Security (TLS) that will further fortify the IP Telephony deployment. TLS encryption makes it almost impossible for an attacker to spoof packets, can provide privacy,
preventing attackers from eavesdropping on signaling, and strong authentication, preventing attackers from tricking SIP proxies into accepting malicious packet floods. Since TLS is not an end-to-end protocol it must be specifically employed in all connections between SIP endpoints. Again, if one link in the chain does not use TLS, the communication is exposed. Unfortunately some popular VoIP platforms do not support SIP over TCP, let alone the use of TLS, like the open-source Asterisk. In addition to the countermeasures already suggested, a VoIP network administrator should also set the SIP Proxies to demand authentication before accepting SIP requests. As already mentioned in the introductory chapters, SIP offers challenge-response based authentication mechanisms for the task. Especially REGISTER requests, which are critical, should require authentication. Authentication for INVITE requests could also be enabled but this might cause problems in the case of large-scale VoIP networks since an external user’s proxy might not have the credentials required to initiate a call with a local user. A security-through-obscurity countermeasure would be to change the port SIP uses (from the default 5060 to something random), but this cannot be considered anything more than an extra layer of security. The use of VoIP-aware Firewall and IPS/IDS (also referred to as VIPS/VIDS if VoIP-aware) can also help significantly since they are usually able to detect malicious SIP traffic. Of course, the deployment of such network devices should not represent a single point of failure for the VoIP network ([115]).

5.2.2 SIP Phone Attacks

As mentioned in the previous section, most of the attacks described above can affect SIP Phones as well - just as long as the target address included in the flood of requests the attacker launches is a valid one. IP Phones also have the disadvantage of limited processing power and weaker security which makes them more vulnerable to such attacks. Moreover, some SIP phone attacks might prove exhaustive for the victimized phone’s battery, thus disabling it until the battery is replaced (see [76]).

**SIP Phone UDP (or TCP SYN) Flood**

The mechanics are exactly the same as in the case of a SIP Proxy being a target. Using udpflood an attacker can flood the target phone and significantly degrade the QoS or render the device unusable. In addition to targeting the SIP port and attacker can direct the UDP flow to the media port (the RTP port) which can either be static or dynamic depending on the phone vendor but can always be determined by sniffing legitimate INVITE requests of the target phone prior to the attack, launching what could be considered an RTP Flood attack. In case the phone runs on TCP, a TCP SYN flood can be used instead. The impact of these attacks varies, depending on who the device vendor is.

**SIP Phone INVITE Flood**

Again there is no significant difference between INVITE flooding a SIP Proxy and INVITE flooding a SIP Phone. The methods described in the previous section can be used to target a SIP Phone as well. Once more, the impact of such attacks varies from one phone vendor to another and ranges from degradation of service to denial of service.

**SIP Phone User Harassment**

Using the inviteflood a hacker can make one or more phones ring at set intervals or even non-stop. While this does not affect the SIP service and does not disable the phones, it effectively makes them useless and can also be considered a harassment of the service’s users. The

---

6 URL: http://www.asterisk.org/
The aforementioned tool could be set by the hacker to send a large number of INVITE requests with a set interval of 20 or 30 seconds between each request. This would frustrate, to say the least, the owner of the victimized IP phone since, each time, shortly after she answers and hangs up the phone, it would ring again. This type of attack could of course be automated to affect many more SIP phones, affecting the availability of the IP telephony service and also the temper and productivity of many of the organization’s employees. Moreover, because of the slow rate at which the requests are sent, most IDS/IPS and other network security devices will fail to detect such an attack, adding to the fact that the INVITE requests can be sent directly to the target SIP Phone, bypassing the Proxy and thus leaving few clues about what goes wrong.

Countermeasures for these types of attacks on SIP Phones are more or less common with the security mechanisms that should be employed to mitigate threats on SIP Proxy server, and which have been discussed extensively in the previous section.

5.2.3 Media Gateway Attacks

Media Gateways are present in all VoIP deployments that need to support analog devices such as traditional analog phones or fax machines, as well as in cases where communication of the IP Telephony service with the PSTN network is required.

Depending on the configuration, some Media Gateways might not have signaling interfaces accessible from the local network because it is not necessary to exchange signaling messages with the IP phones directly. What is necessary is to have the media interfaces of the Gateways open to the local VoIP network because the media traffic (voice) must flow from the phone to the Gateway, where it will be translated and forwarded, if external calls are to be made.

In case there is a SIP signaling interface on the Gateway that the attacker can access from the local network, then she would be able to launch any of the attacks that she would launch against a SIP Phone or Proxy (e.g. use the udpflood). Additionally, with some sniffing of media sessions, she would probably manage to locate the RTP port that the Media Gateway will certainly have open for communicating with SIP Phones. All she would have to do next is target the UDP flood tool on that port. Both these attacks will certainly consume Gateway resources and could even succeed in preventing all inbound and outbound calls from getting through. A slightly modified version of the udpflood tool, the rtpflood7 utility, is better suited for the task.

Other than the tools already mentioned, hackers have a variety of SIP-oriented flooding tools in their arsenal, such as SIPP8, SIPSak9 and even the SIP Forum Test Framework10.

In summary, SIP Phones, SIP Proxies and Media Gateways are extremely vulnerable to flood attacks - especially if run over UDP. The attacker need not be especially skilled to bring a VoIP deployment to its knees, unless the network is properly secured. Countermeasures exist but must be implemented across the entire network, since, especially in this case, the security is as strong as its weakest.

---

7URL: http://www.hackingvoip.com/tools/rtpflood.tar.gz
8URL: http://sipp.sourceforge.net/
9URL: http://sipsak.org/
5.3 Signaling and Media Manipulation

Other than flooding or sending malformed packets to her target, thus causing a denial or disruption of service, an attacker can also manipulate the signaling and media traffic, launching a range of possibly more sophisticated and stealthy attacks.

5.3.1 Registration Removal

SIP phones send a REGISTER Requests to the Registrar Server on bootup to register its location. The Registrar processes the request and updates the information on the Location Server, which is queried by the Proxy Server so that the latter knows where to direct incoming calls. As already mentioned in the introductory chapters, all of the above services are usually run on the SIP Proxy. After bootup, SIP phones refresh their registration by sending new Register requests. This is done at fixed intervals and depends on the phone vendor (most SIP phones re-register every one hour by default) but a new interval can be set by the SIP proxy and the phones configure themselves accordingly. A normal User Agent registration can be seen in 5.1.

In the case that an attacker manages to remove a phone’s registration then the Proxy will not be able to locate the phone through the Location service and thus it won’t be able to direct incoming calls to the phone. In other words, the phone will not be able to receive any calls. This does not affect the ability of the victimized phone to make calls. Registration can be removed if a REGISTER request is sent containing a "*" in the "Contact:" header field and a zero in the "Expires:" field. This attack can be automated using the erase_registrations¹¹ utility to erase all SIP phone registrations. Moreover, an attacker could script this attack, so that the

¹¹URL: http://www.hackingvoip.com/tools/erase_registrations.tar.gz
tool re-runs after short intervals, to make sure it will continue to block the victimized phone even if it automatically re-registers. This could assure the attacker that the target phone will not receive any calls for as long as the script runs. Another tool that could be used for this and many more attacks is SiVuS\textsuperscript{12} VoIP vulnerability scanner.

The impact of these attacks can be high since one or more phones will not be receiving any calls. If an attacker succeeds in attacking the phone of an executive or, even worse, the customer support line or phone-orders line, the attack will certainly affect business in an unacceptable way. Of course, once more, for these attacks to take place the attacker needs to have access to the internal network but this is far from impossible, as described in previous chapters. This is especially true nowadays with the widespread use of rootkits, viruses, backdoors etc by hackers. The need for an effective and enforced corporate-wide security policy is obvious.

The countermeasure against Registration Removal attacks are more or less the same to the ones suggested for SIP phone and SIP proxy attacks in the previous section. VLANs and the use of TCP with TLS, enabling authentication for REGISTER requests, using SIP firewalls and changing default service port numbers are essential. Additionally, decreasing the re-registration interval on the SIP phones can help but, still, if the attacker decides to script the attack and remove registrations every few seconds, this will not be of much use.

5.3.2 Registration Addition

A user of the IP telephony service can register many locations under his contact name in the Registrar server. This allows someone to have all his/her phones (office, lab etc) ring once there is an incoming call.

This feature can be exploited by a hacker in a few ways. A utility that can be used to add new contacts under an existing username is add\_registrations\textsuperscript{13}. An attacker could use this tool to register all phones extensions (or at least all the phone extensions she knows) under a single username. This would mean that once someone calls the target phone, all the aforementioned phones will ring. With a little bit of scripting it would not be hard to do this for all the known service users. This would lead to the extremely frustrating situation where whenever anyone calls anyone else, all the phones will be ringing. Certainly not a pleasant situation which, apart from being frustrating, essentially renders the IP telephony service useless.

In a variation of this attack a malicious person could use the add\_registrations or SiVuS tools to add a compromised SIP phone under the victim’s username. This is essentially a basic form of a registration hijack. Whenever someone calls the victim, the hacker’s phone rings as well and she will probably have the time to pick it up before the victim. This opens the way to various phishing and other attacks that will be discussed later on.

The countermeasures for this type of attacks are identical to the ones that should be deployed to protect from Registration Removal attacks.

5.3.3 Registration Hijacking

A basic Registration Hijacking has been mentioned in the previous section. In that case the attacker just adds one more phone extension under the victims contact card. In a full-scale

\textsuperscript{12}URL: http://www.securityfocus.com/tools/3528
\textsuperscript{13}URL: http://www.hackingvoip.com/tools/add\_registrations.tar.gz
registration hijack the attacker goes one step further and removes the original registration - using the way described in the first attack of this section or through a DoS attack or otherwise - before proceeding with inserting a registration with her own SIP Phone. Figure 5.2, taken from [96], demonstrates such an attack, in which the attacker chooses to start off with a DoS attack to disable the victim’s phone.

Reg hijacker\textsuperscript{14} is a tool that can be used to hijack user registrations and which basically automates the steps described above (removes all contacts under the target username and adds a new one with the malicious phone’s address) and has the ability to operate in VoIP networks that authenticate Register Requests. This tool can be a extremely dangerous in the wrong hands. It opens up many attack options for hackers. The simplest attack would be to hijack a user’s registration and redirect all calls to a non-existent SIP phone so that the victim receives no calls. Moreover it could be used to swap registrations between two phones which would lead to an analogous swap of the incoming calls; one more frustrating situation for the two victims. Moving to more sophisticated attacks, the registration hijacking technique could be used to perform an application layer man-in-the-middle attack where the attacker will be using a rogue application that acts like a SIP phone and which manipulates all signaling and media before allowing the traffic to go through to its original destination. It could also be used for phishing attacks, discussed in more detail later. In the case of a large-scale VoIP deployment it could mean that the customers calling a phone-banking service could have their calls re-routed to the hacker’s SIP phone, which plays exactly the same pre-recorded messages as the bank’s phone service. This is obviously a critical situation.

\textsuperscript{14}URL: http://www.hackingvoip.com/tools/reghijacker.tar.gz
The caused by Registration Hijack attacks can be substantial for a victimized organization. Basically, the hacker’s imagination is the limit to the attacks that could be launched. All SIP phones could be registered under the CEO’s username, causing his phone to ring constantly and depriving everyone of the IP telephony service. No organization would think of relying on VoIP when there is a chance that the phone-sales department calls are routed to a non-existent line or, even worse, to a competitor’s phone-sales service (in the case of public VoIP networks).

Countermeasures are no different to the ones suggested for previously mentioned signaling manipulation attacks.

5.3.4 Redirection Attacks

In the SIP Responses specified by the IETF in RFC 3261 (see [49]), a hacker might find three responses particularly interesting (copying directly from the aforementioned RFC):

301 Moved Permanently
The user can no longer be found at the address in the Request-URI, and the requesting client SHOULD retry at the new address given by the Contact header field.

302 Moved Temporarily
The requesting client SHOULD retry the request at the new address(es) given by the Contact header field. The Request-URI of the new request uses the value of the Contact header field in the
response.

305 Use Proxy
The requested resource MUST be accessed through the proxy given by the Contact field. The Contact field gives the URI of the proxy. The recipient is expected to repeat this single request via the proxy.

The 301 Moved Permanently, 302 Moved Temporarily and 305 Use Proxy are sent by SIP proxies and phones (the 305 response can only sent by phones and other user agents) as a response to INVITE requests in the situations specified in the above RFC quote. There is nothing to stop an attacker who has access to the VoIP network from sniffing traffic and, as soon as she spots an INVITE request, to reply with a 301, 302 or 305 response thus redirecting the victim to the SIP Phone - or SIP proxy in the case of 305 - of her choice, the address of which she includes in the "Contact:" header of the response.

The call could be redirected to: a non-existent phone/proxy, causing a DoS for the caller and the called party; to a random phone, causing a DoS to the called party who can’t receive the incoming calls and mostly frustration to the caller and the user she was redirected to; to a rogue SIP phone/proxy or a rogue application acting as a SIP phone/proxy which would enable other threats such as MITM and/or phishing attacks. Redirectpoison\(^\text{15}\) is a utility that implements this attack and which features a lot of customization options. Figure 5.3, taken from [4], shows a 305 Use Proxy Redirection Attack.

Redirection Attacks are similar to Registration Hijacking attacks described above because both are used to subvert a legitimate call to another SIP phone of the hacker’s choice. Their main difference is that the Redirection technique is a real-time attack (requires immediate response once the INVITE Request is detected) that cannot be automated effectively - unless

\(^{15}\text{URL: http://www.hackingvoip.com/tools/redirectpoison.v1.1.tar.gz}\)
perhaps the hacker is really dedicated - and it can only affect one communication at a time, while Hijacking can be used to divert many calls at once, current and future ones.

Once more, countermeasures are common to those suggested for protecting against all other signaling manipulation attacks.

5.3.5 Session Teardown

It should be reminded that a **BYE Request** is sent from one SIP phone to another to announce the termination of their call in progress. This request might be routed through the SIP Proxy or sent directly from phone to phone, depending on the VoIP network configuration. The implementation most commonly chosen in a corporate environment is to route requests through the Proxy. This way the server can maintain call state which is necessary if advanced call features (e.g. call accounting) are to be used.

What makes things more interesting for attackers is that, in both cases, the SIP phones will process **BYE Requests** normally and act accordingly. Using tools such as sip-kill\(^{16}\), sip-proxykill\(^{17}\) and teardown\(^{18}\), a malicious user can send BYE Requests to target phones and force them to terminate the call in progress. It does require some network sniffing prior to the attack to get the necessary data from the SIP requests and responses exchanged during the call setup, but this is trivial, as already mentioned. This attack can either be launched against the SIP proxy (which will forward the BYE request to the target phone) or directly against the target SIP phone. In both cases it is 100% effective and does not depend on who the target phone’s vendor since it relies on basic SIP call teardown mechanics that are definitely supported by all SIP devices.

Countermeasures are common with the rest of the attacks discussed in this chapter and include the use of TCP and TLS, the use of VLANs to segregate voice and data networks, the use of SIP Firewalls and generally VoIP-aware security devices as well as the use of non-default ports for VoIP services - the latter only as one more layer of security since security-through-obscurity cannot be considered an effective countermeasure. Enabling digest authentication for Requests can help. In specific, using authentication for BYE requests can protect SIP proxies from Teardown attacks but, obviously, strong passwords are needed. If an attacker can guess the weak passwords users choose then authentication will not offer anything more than overhead. It should be noted that Request authentication cannot protect SIP phones from direct Teardown attacks.

5.3.6 SIP Phone Reboot

The IETF have defined in RFC 3265 (see [50]) specific event notifications that are necessary in order to offer increased functionality to SIP deployments. A flashing envelope on the LCD screen of the IP phone when the user has new voicemail messages is an example of such a feature that requires the use of the aforementioned extensions. In a way, the SIP phone must subscribe to these features in order to be able to use them. This is done by sending a **SUBSCRIBE** request to the SIP proxy which then will send a **NOTIFY** request when needed (e.g. whenever there is new voicemail in the case of the previous example).

---

\(^{16}\)URL: http://skora.net/uploads/media/sip-kill

\(^{17}\)URL: http://skora.net/uploads/media/sip-proxykill

\(^{18}\)URL: http://www.hackingvoip.com/tools/teardown.tar.gz
This would not be of an issue for VoIP security if vendors had programmed their devices not to process NOTIFY requests unless there was a SUBSCRIBE request to the event in the past. Unfortunately, this is not the case. Some phones process NOTIFY requests for event that they never subscribed to and this can be a real threat to the availability of the IP telephony service since some requests can severely affect the functionality of the SIP phones. The check-sync request can force some phones to reboot. The SiVuS tool mentioned earlier can also be used for this type of attacks.

Again, countermeasures are no different to the ones suggested in previous sections of this chapter.

### 5.3.7 Caller ID Spoofing

The Caller ID in SIP is based on the information found in the "From:" header field of various requests. Unfortunately, it is extremely easy for an attacker to spoof this field and thus spoof the Caller ID.

There are plenty of VoIP security tools, most made for launching various other attacks already discussed, and which allow, as an option, to define the contents of the "From:" field. One example is SiVuS. Moreover, there are companies dedicated in providing Caller ID spoofing services, like SpoofTel.

Arguably, Caller ID spoofing is not exactly an attack on its own but it is used as a complementary manipulation of the signaling protocols to help make other attacks more effective. More on that, as well as on countermeasures for Caller ID spoofing, in the Social Engineering Threats chapter. Still, Caller ID spoofing can be used for toll fraud in deployments where the "From:" field information is used to charge users for their calls. This should, obviously, be avoided.

### 5.3.8 RTP Insertion & Mixing

As already pointed out in Chapter 2, RTP is the media protocol of choice for virtually all VoIP deployments, no matter what the signaling protocol used is - which makes attacks on RTP a threat for all VoIP networks worldwide. RTP runs over UDP since TCP is not really an option. In the case of real-time transmissions of data, such as voice, TCP adds quite a lot of overhead without any serious advantages. The retransmission of lost packets is not of any use for this type of application since by the time the packets are retransmitted the conversation will have moved on and the retransmitted packets would probably do more harm than good if they were processed.

The codec used defines the timeframe over which audio is sampled and the transmission rate of RTP packets. Typically, the G.711 codec is used in VoIP deployments. Although the transmission rate is fixed, the rate at which the packets are received at the other endpoint is not. UDP is a connectionless protocol and thus the RTP packets might arrive in a different order to the one sent or might not arrive at all, depending on the network health and traffic. Higher layers try to compensate for the weaknesses of UDP, building on the information provided by the RTP header (Sequence Number, Timestamp and SSRC - see Section 2.3), and it is usually accomplished with the use of a Jitter Buffer.

---

19 URL: http://www.spooftel.net
Manipulating the Media traffic is not a very easy task. The attacker must spoof the RTP header data in order to have any chance of tricking the target into accepting her malicious traffic. If this is not done - or not done properly - the malicious traffic will simply be rejected and the legitimate conversation will go on unharmed.

One possible attack to the RTP media is to insert new audio in a conversation, which means that the legitimate audio will be replaced (if everything is done right of course). Another type of attack involves mixing new audio into the audio stream. This means that the injected audio will be processed along with legitimate packets and the victims will hear a background noise or some other sound along with their voices. Two tools that can be used to launch the aforementioned attacks are \texttt{rtpinsertsound} and \texttt{rtpmixsound}, respectively. It must be apparent that these tools need to be run from a host that gives them the ability to monitor the traffic to be attacked. This is the only way they can gather the necessary information that will allow them to effectively spoof the RTP headers of the malicious traffic.

It should be pointed out that with these tools only affect one side of the call. The other side will not hear any of the inserted or mixed audio and will not be aware of the attack because every instance of these utilities can only affect either the upstream or the downstream. Of course at some point the affected side will start complaining to the other about the audio artifacts or whatever it is she is listening. In the case of injected audio, though, even if the victim starts complaining she will not be able to hear the answer of the other party as she will only be hearing the recorded audio that the attacker inserted.

The malicious person tampering with the RTP traffic could always launch two instances of the RTP injection or mixing application and target each one to a different side, thus affecting both sides. She could also launch multiple instances of the utilities to disrupt many calls at the same time. There are numerous attacks an attacker can launch since, in essence, she is in control of the media stream. One possibility would be to insert background noise significantly degrading the QoS. Another attack could involve inserting offensive language into the call to give the impression to the victim that he is being insulted and abused. The latter could be an extremely detrimental attack in the case of a customer calling the technical support line of the target organisation and experiencing this offensive behavior. Once more, the only limit is the hacker’s imagination and determination.

Again it should be noted that one of the most important advantages of these attacks is that they can be launched against any VoIP deployment, be it SIP or H.323 or even proprietary ones, since everyone uses RTP as the media protocol. Using the above techniques an attacker can seriously undermine the credibility of organizations, annoy or insult individuals and degrade the QoS of the VoIP network.

One obvious countermeasure that can be employed to mitigate media manipulation attacks is audio encryption. \textit{SRTP} can provide the required encryption and authentication of the media stream. Encryption alone will of course provide privacy, preventing mixing attacks, but it will not prevent media insertion attacks. Still, even if voice is inserted, the malicious media will not be processed properly as it will not be encrypted and thus the only thing an attacker can achieve is to generate noise - which means that more sophisticated attacks are not an option anymore. SRTP provides the option to also use authentication at the endpoints, which can help the endpoints recognise malicious traffic and discard it. SRTP is supported by most vendors but, unfortunately, it is still rarely used. \textit{ZRTP} is another protocol - a more secure extension

\url{http://www.hackingvoip.com/tools/rtpinsertsound_v3.0.tar.gz}
\url{http://www.hackingvoip.com/tools/rtpmixsound_v3.0.tar.gz}
of SRTP as already explained - that could be used for the task.

The use of VLANs can help against this type of attacks as well since the logical isolation of voice traffic from standard data traffic can make it harder for an attacker to monitor the media stream. It essential to note that the use of Softphones can compromise the security provided by VLANs - but more on that in the relevant chapter.

In addition to the above VoIP-aware Firewalls and IPS/IDS devices should be deployed that offer monitoring of RTP and can detect the injection of malicious traffic. Such solutions include, but are not limited to, SecureLogix\textsuperscript{22} VoIP Firewall for Hybrid networks, Borderware’s\textsuperscript{23} SIPassure and TippingPoint’s\textsuperscript{24} UnityOne. What should be mentioned is that, based on tests conducted for research purposes such as [85], in reality all of these solution are still under heavy development and most are not in a position to offer 100% protection, no matter what their vendors might say. On the positive side, as VoIP adoption increases, the security researchers focus more on the subject and the security mechanisms for protecting VoIP deployments mature.

\textsuperscript{22}\textbf{URL}: http://www.securelogix.com/
\textsuperscript{23}\textbf{URL}: http://www.borderware.com/
\textsuperscript{24}\textbf{URL}: http://www.tippingpoint.com/
Chapter 6

Non-SIP-Based Deployments, A Brief View

The fact that this Thesis is mostly focused on SIP-based VoIP deployments should not give the impression that other signaling protocols are more secure. To begin with, the RTP vulnerabilities are obviously common to all implementations of VoIP, regardless of the signaling protocol used, as its the media protocol of choice for all vendors.

6.1 H.323 Protocol

H.323, although older than SIP and perhaps more mature, has some inherent security issues. One important issue with H.323 deployments is that many of the protocols under the H.323 umbrella use random ports making it extremely complicated to secure through firewalls ([119]). As the ports used by H.323 are not set the firewalls should be configured to keep all possible ports open. This is not an acceptable security policy because that would mean leaving 10,000 UDP ports open as well as several TCP ports. An H.323-aware firewall is the only way to ensure signaling traffic will go through these ports while all other traffic will be blocked. This gives rise to one additional problem. The complex nature of H.323 traffic makes it hard for H.323-aware firewalls to analyze and, thus, the time needed to parse the traffic introduces extra latency in the communication ([75]). Network Address Translation (NAT - [109]) is also a problem for H.323 because the IP address and port listed in the IP header are not the same as the internal addresses specified in the H.323 header. The only option is to have a NAT gateway that can read the H.323 messages and translate the addresses. This processing can, again, affect QoS.

Moreover, and as already mentioned in the Fuzzing section, the PROTOS project has discovered vulnerabilities in H.323 (in the H.225 protocol in specific - see the relevant CERT Advisory [11]) that allow denial of service and even execution of code attacks to be launched if exploited. The latter attack, which is also the most critical, is possible because of the insufficient bound checking done when systems parse and process the H.225 messages.

Of course this is just the tip of the iceberg since there are yet many to be discovered when every protocol in the H.323 family is thoroughly examined through fuzzing or other techniques ([114]). This is because of the numerous different vendor implementations that exist, the inherent complexity of this protocol family, the problematic implementations of the encoding/decoding algorithms and the fact that the H.323 protocols, each on its one and as a whole, have not endured the same degree of security testing that other, more common protocols have ([30]).
Most utilities presented throughout this Thesis that could be used to attack a SIP deployment one way or another, can either be used against H.323 deployments as well or equivalent tools exist for the same tasks.
6.2 MGCP Protocol

Finally, MGCP, just like SIP, is vulnerable to various signaling manipulation attacks. A malicious user could eavesdrop on conversations (even conference calls), cause a Denial or Disruption of Service, call diversions, call disruptions, perform a MITM attack and so on, as demonstrated in [113]. Figure 6.1 demonstrates an eavesdropping attack through MGCP signaling manipulation. A significant difference that increases risk of MGCP manipulation is that, typically, MGCP communications occur across public and generally not safe networks like the Internet ([25]). The use of IPSec is recommended in [3] to provide encryption and authentication between Gateways.

![Eavesdropping with MGCP](image)

Figure 6.1: Eavesdropping with MGCP ([113])
6.3 Cisco Skinny Call Control Protocol

When it comes to proprietary protocol implementations things are not any better. Cisco’s\(^1\) Skinny Call Control Protocol (SCCP) is quite popular for deployments that utilize the Cisco Unified Communications Manager (CallManager). Searching Cisco’s security advisories using “sccp +vulnerability +advisory” as keywords returns 57 results which are all vulnerabilities in the protocol or the underlying platform that pose a threat, one way or another, to the confidentiality, integrity and availability of an SCCP-based IP telephony service. The results are comparable when searching advisories published by organizations such as NIST and CERT. Some examples, quoting the relevant advisories:

- **SCCP/SCCPS Port Scan Denial of Service Vulnerability** ([16])
- **IP Phone with SCCP firmware may be susceptible to crafted ICMP ”hard” error messages.** ([17])
- **Certain malformed packets sent to the SCCP port on an IOS device configured for ITS, CME or SRST may cause the target device to reload.** ([18])
- **Integer overflow in the get_input function in the Skinny channel driver, as used by Cisco SCCP phones, allows remote attackers to execute arbitrary code.** ([93])

\(^1\)URL: http://www.cisco.com
Chapter 7

Softphones

Softphones are VoIP applications that, when run on a computer, allow the user to make phone calls. In other words, a softphone is a software application that emulates a VoIP phone.

In order to make calls a microphone and a set of speakers or, alternatively, a headset must be connected to the computer. There is also a market of regular-looking phones that are intended for use with a softphone application. These devices are usually connected to the PC that the softphone runs on through the USB interface, although, nowadays, wireless models are not uncommon.

Softphones are an appealing choice for many enterprise VoIP deployments mostly because the cost of buying and installing softphone applications is negligible at least when compared to the cost of buying and installing physical handsets). This is not the only benefit though. Softphones clearly have an advantage when trying to accommodate the needs of staff who are on the move a lot. People that work on the road most of the day visiting clients or people who work at home can connect to the enterprise VoIP network as long as they have their laptop and an Internet connection. Moreover, softphones can offer more features than regular handsets such as instant messaging, LDAP and email integration. In other words, with softphones it is trivial to bridge voice and data VLANs.

Therefore, it should not come as a surprise that most of the vendors that dominate the enterprise VoIP market offer softphone solutions. Some examples are Cisco’s IP Communicator¹, Nortel’s IP Softphone 2050², 3Com’s NBX pcXset Softphone³ and Avaya’s IP Softphone⁴ (see Figure 7.1)

²URL: http://products.nortel.com/go/product_content.jsp?parId=0&segId=0&prod_id=24043
³URL: http://www.nbxsoftware.com/nbxphones/nbxpcxset.html
Of course, the available types of softphone applications are not limited to enterprise solutions as the ones mentioned above. Softphones also include applications which are developed for public use and route calls via the Internet. Skype\textsuperscript{5} is the most popular application in this category, boasting over 171 million users worldwide and over 500 million downloads. The Gizmo Project\textsuperscript{6} is another popular choice with a user base of over 1 million users. Softphone functionality is also becoming commonplace for instant messaging applications that extend their features to support VoIP services and some of the most popular examples of this trend are Microsoft Live Messenger\textsuperscript{7}, Yahoo Instant Messenger\textsuperscript{8} and Google Talk\textsuperscript{9}.

Although the aforementioned applications are typically not utilized - or at least not officially implemented and supported - in enterprise environments, it is highly likely that these softphone technologies, perhaps with some variations in the client software, will enter the enterprise market soon enough.

\textsuperscript{5}URL: http://www.skype.com/
\textsuperscript{6}URL: http://gizmoproject.com/
\textsuperscript{7}URL: http://get.live.com/messenger/overview
\textsuperscript{8}URL: http://messenger.yahoo.com/
\textsuperscript{9}URL: http://www.google.com/talk/
7.1 Softphone Security

Softphone applications can be considered inherently insecure since most of them rely in Peer-to-Peer or instant messaging engines (or a hybrid of both) and carry the security vulnerabilities these engines demonstrate.

It has already been mentioned that VoIP security relies heavily on the security of the underlying platform the VoIP service is running on. This includes operating system vulnerabilities, service vulnerabilities, firmware vulnerabilities and so on. Consequently, softphone security is largely affected by the security state of the computer it is installed on. This is definitely not a positive situation for softphone security since compromising a desktop computer has become a trivial task nowadays. Worms, rootkits and other tools are easily found on the Internet and can utilize sophisticated and complex mechanisms for the purpose of compromising a target host while being very easy to use even for someone with little or no network and hacking knowledge. In other words, and without getting into a detailed enumeration of the ways a personal computer can be compromised nowadays, it is a given that the entry points a hacker could possibly find on a desktop PC are a lot more than the entry points on an IP phone or any other device with limited functionality and thus lower exposure surface. If a hacker or script-kiddie compromises a user’s computer then eavesdropping on all traffic, including VoIP conversations, is the least of the things she can do.

Additionally, softphones, in order to operate, require that a number of ports are open on the host systems and on the firewall that system is protected from. These open ports pose a security threat. The user might install a Peer-to-Peer (P2P) file-sharing program that will communicate with other hosts via the softphone ports to download infected software or other files, exposing the softphone host, the corporate network and even the corporation to numerous dangers such as confidential information disclosure or even a regulatory breach (since the P2P program could be used to exchange illegally downloaded music or even child pornography material). Moreover if, one way or another, the system is infected by a worm or a virus or any other malicious program then that program could potentially use the softphone ports to establish a reverse connection to the attacker’s computer.

One of the biggest security issues with softphones is a direct consequence of their main feature which is the easy way they manage to bridge voice and data VLANs. The defense-in-depth security model suggested for VoIP deployments throughout this Thesis is largely based on the logical segregation of voice and data networks through the use of VLANs. In such a network, a corporate computer with a softphone application running on it can be considered a link between the voice and data VLANs. The reason is simple: If an attacker compromises this host through the data VLAN then he can exploit the VoIP capability of the softphone application to launch attacks against the voice VLAN. In other words, the extra security VLANs introduce is essentially canceled out by the use of softphones. The aforementioned risk is magnified in the case of Wireless VoIP phones since they can indeed act as a stepping stone for an attacker to hop from one VLAN to the other, while, at the same time, they carry the inherent security issues of wireless networking such as the ease of eavesdropping (unless strong encryption is utilized).
7.2 Skype

Skype is by far the most popular softphone in use today so it is necessary to examine it in more
detail. Skype is largely based on the KaZaA\textsuperscript{10} file-sharing utility since the developers of the
two applications are the same. So, Skype is using a proprietary P2P VoIP network to route
calls between its users. It features a SkypeIn service that assigns a phone number to each user
so that the Skype client can receive calls from any phone and also a SkypeOut service which
allows Skype users to call PSTN phone numbers.

In October 2005 Skype was bought out by eBay\textsuperscript{11} in an 1.4 billion pounds deal\textsuperscript{12} and,
although everyone predicts that sooner or later it will be made available as a robust enterprise
solution, at the moment such a product has not been launched. Skype, being a very popular
application, is of course used in the enterprise sector but mostly by employees for their own
personal needs.

![Skype call in progress](image)

Figure 7.2: A Skype call in progress.

\textsuperscript{10}URL: http://www.kazaa.com
\textsuperscript{11}URL: http://www.ebay.com
\textsuperscript{12}URL: http://news.bbc.co.uk/1/hi/business/4237338.stm
The developers of Skype, in the official website, describe their network as a decentralized P2P network which they call Global Index and which they describe as: "a multi-tiered network where supernodes communicate in such a way that every node in the network has full knowledge of all available users and resources with minimal latency". The developers chose to utilize variants of STUN (which stands for Simple Traversal of UDP through NATs - defined in [102]) and TURN (which stands for Traversal Using Relay NAT - see [100]) to allow communication even behind firewalled NATs.

Skype is very effective in initiating calls even in situations where strict network security policies are in place. In the first step of a Skype call the caller tries to contact the callee directly and if this fails then the callee tries to contact the caller instead. If none of the direct connections are feasible then the two parties resort to a third skype user who is reachable by both hosts and who acts as a supernode and routes the call for them. Any skype user can act as a supernode if needed - this is stated clearly in the Skype privacy agreement the users "sign" during Skype’s installation - but, it should be noted that this can cause higher (potentially double according to [36] bandwidth consumption compared to the standard client operation.

7.2.1 Skype Security

The developers of Skype assert they have worked a lot on providing call confidentiality and integrity. The security mechanisms implemented in Skype are proprietary and thus very little is known about them. Still, in October 2005, an independent review of the encryption algorithms used in Skype was conducted and concluded that:

"The designers of Skype did not hesitate to employ cryptography widely and well in order to establish a foundation of trust, authenticity, and confidentiality for their peer-to-peer services. The implementers of Skype implemented the cryptographic functions correctly and efficiently. As a result, the confidentiality of a Skype session is far greater than that offered by a wired or wireless telephone call or by email and email attachments".

More information on the cryptography utilized by Skype and the tests conducted can be found in the the Review report (see [6]).

Skype has had a number of vulnerabilities exposed from independent security researchers. At the moment, in the security bulletins section of Skype’s website a total of seven vulnerabilities are listed. These vulnerabilities have now been patched of course but it is highly unlikely these are the only ones. It should also be noted that researchers have managed to reverse-engineer Skype and provided an in-depth analysis of the applications (which included information about Skype’s cryptography scheme and the discovery of a vulnerability) at the RECON Reverse Engineering Conference, in June 2006. For more information see [27] and [28].

Research ([54]) has shown that Skype is not as aggressive in its use of network bandwidth as most P2P applications. Still, a widespread use in a corporate environment could cause bandwidth congestion and generally degrade network health. And, nowadays, its not just about softphones installed on the organization’s personal computers. Employees installing Skype on their mobile phone and use the phone’s WLAN capability to place calls through the corporate network - a realistic scenario with new generation mobile phones - can easily go unnoticed while consuming critical network resources. These situations would not be a problem if rate-shaping technologies typically available in a corporate network were effective at detecting Skype traffic,
but, unfortunately for network administrators, they are not. Limiting the amount of bandwidth Skype consumes - or even blocking it altogether - can be a difficult goal to achieve since it encrypts and obfuscates traffic its traffic making it difficult for network devices to recognise it. Various firewall and rate-shaping devices feature, according to their vendors, Skype traffic detection and there is even a dedicated network device for the task, the Aconix L7 Skype Manager v2.0 Security Appliance\textsuperscript{14} but, in fact, these devices have limited effectiveness. Every new major Skype release uses more sophisticated payload obfuscation than the previous one, managing to evade the aforementioned devices.

In a corporate environment, the best way to block Skype is to include a clause in the security policy that prohibits its installation on the company’s computers and, of course, to enforce it.

### 7.3 Recommendations

At this point it must be clear that the use of softphones can expose a corporate network to a number of serious threats. Tracking and controlling the use and network resources utilization of softphones is a hard task to accomplish. It should not come as a surprise that the Defense Information Systems Agency of the U.S. Department of Defense in \[25\], the National Institute of Standards and Technology in \[75\] and security experts(\[5\]) alike strongly discourage the use of softphones. If the use of Softphones can be avoided then they should not be used and this should be reflected in the security policy that will prohibit the installation and use of such applications.

In the cases that the use of softphones cannot be avoided then certain measures have to be taken to mitigate part of the risk. It goes without saying that all the applications must be installed and configured in an organized way by the staff assigned to the task and not by individual users and that the machines the softphone applications are installed on must be patched and up-to-date. Moreover, home-office users that need to use their VoIP softphone at home and generally roaming users pose a serious threat because they will probably use the VoIP service from an uncontrolled environment and thus they must do so via a VPN connection to the corporate network. Their traffic must then be routed to different VLANs depending on if it is voice or data. Finally, according to \[25\], the best way to somewhat mitigate the extra threats introduced by softphones is to extend the defense-in-depth VLAN approach in a way that computers utilizing softphones have a dedicated voice VLAN and a dedicated data VLAN, different to the voice and data VLANs the rest of the hosts belong to.

\textsuperscript{14} URL: \url{http://www.akonix.com/press/releases-details.asp?id=110}
Chapter 8

Social Threats

8.1 SPAM over Internet Telephony (SPIT)

The term SPAM refers to unsolicited bulk email messages. Virtually anyone who has an email address will be aware of these unsolicited email messages that practically bombard users with advertisements of gambling websites, health products, diet pills and mortgages, to name a few common cases. Nowadays an email service user, depending on the exposure of his email address, can get over 100 SPAM messages a day. Figures 8.1 and 8.2 show the most popular subjects of SPAM emails and the countries most SPAM-sending hosts originate from, based on the real-time monitoring tools of the Commtouch Software Online Labs

<table>
<thead>
<tr>
<th>Sexual Enhancers</th>
<th>53.6%</th>
<th>Pornography</th>
<th>2.6%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stock Pump and Dump</td>
<td>19%</td>
<td>Software</td>
<td>1.4%</td>
</tr>
<tr>
<td>Other Pharmaceuticals</td>
<td>12%</td>
<td>Electronics</td>
<td>1%</td>
</tr>
<tr>
<td>Finance</td>
<td>3.8%</td>
<td>Other</td>
<td>6.4%</td>
</tr>
</tbody>
</table>

Figure 8.1: Subjects of SPAM email (First quarter of 2007)

1URL: http://www.commtouch.com/Site/ResearchLab/statistics.asp
As an extension of the above, *SPAM over Internet Telephony* (SPIT) refers to the bulk unsolicited VoIP calls. SPIT can be considered a highly sophisticated, automated and more massive form of telemarketing calls. SPIT calls could become as common as SPAM messages, unlike telemarketing which, although annoying, has a relatively low rate of calls. A few calls per week cannot be compared to the hundreds of SPAM emails one might get in the same timeframe. It is essential to look into the benefits that VoIP technology brings to people who use unsolicited calls to advertise their products - and also to people who make money out of sending unsolicited emails and who now have a new powerful weapon in their arsenal.

### 8.1.1 Why SPIT?

Telemarketing calls are usually only partly automated - but this is only one of the reasons for its low call rate. The most common technique used in well-organized telemarketing corporations is to dial the numbers using an auto-dialer and, if there is an answer, to forward the call to a human who will try to start a discussion on the product. Some sophisticated auto-dialers can even tell if the voice that answers the call is a pre-recorded message from a voicemail system/answering machine or a human talking real-time. The reason why these operations are only partly automated is that the cost to setup a telemarketing operation that will be able to support numerous PSTN calls is significant, the cost of the calls themselves can be significant as well and since a pre-recorded message cannot offer the same sales rate that a human can achieve - especially if the latter is a good salesman - choosing to use a fully-automated deployment would essentially undermine their investment. The investment is far from negligible since the cost for the PBXs, the phone lines, the calls themselves (especially in cases of long-distance calls) and, of course, the salary of the people hired to handle the calls, all of which are required for a large-scale telemarketing operation, can easily reach tens of thousands of pounds for the initial investment and significant monthly expenses. Taking into consideration that the vast majority of the calls will not be answered or will be answered by an answering machine and that only a small percentage of the calls that actually go through to a potential client will result in a sale, it is easy to understand why such business ventures are considered high-risk and inefficient compared to the investments required.
The requirements of telemarketing over the PSTN described above explain why this type of “phone SPAM” is not as big an issue as email SPAM. Unfortunately for consumers, SPIT is for telemarketing what email SPAM is for traditional mail advertisements. SPAM is completely free for the attacker while printing out and sending brochures by post requires money and tracing back the sender can be relatively easy, which largely explains why people can find over 100 SPAM messages in their inbox per day and only maybe 5-6 brochures per week on their doorstep. Respectively, while telemarketing involves a significant long-term investment the entry into the SPIT business can be very economical.

For one, since the call rate is way higher with SPIT the aimed sales-rate can be reduced, allowing the use of fully automated procedures and eliminating the need to employ humans. Moreover, while traditional PBXs are very expensive devices (especially in cases where a lot of simultaneous calls must be supported), VoIP PBXs can come completely free. The open source Asterisk® IP PBX is a robust solution that allows the setup of a feature-rich PBX with the only cost being that of the computer it will be installed on. The use of softphones eliminates the need to buy handsets and, of course, high-speed network access eliminates the need of having to pay for numerous PSTN phone lines. As described in [36], in the scenario of a broadband line which offers an 1.5MB bandwidth to the attacker, she would be able to launch 150 call attempts per second (assuming a SIP INVITE message of 1K packet size) and, using the G.711 codec, her Internet connection would be able to support 20 simultaneous calls. If codecs that offer more compression, like G.729, are utilized then up to 100 simultaneous calls could be supported over that single 1.5MB connection.

Another area where SPIT clearly has an advantage over PSTN-based telemarketing is that of call charges. While at the moment most VoIP calls will terminate on the PSTN network, thus resulting in similar charges to a normal PSTN call, when a call does in fact terminate on another VoIP network the costs are significantly lower. As more and more of these calls end up being direct VoIP to VoIP connections this cost will be extremely low or even free. Even today, and as an indication of things to come, various vendors offer Softphone applications that feature completely free VoIP-to-PSTN calls for selected destinations around the world and very low rates for all other destinations and even VoIP-to-Mobile calls (e.g. VoIPBuster®, FreeCall®, VoIPDiscount®). Integrating the aforementioned applications in a SPIT scheme will enable an attacker to minimize her operating costs.

Combining all of the above with an “anonymizing” service, typically available for free on the Internet, she can also ensure that her scheme’s bulk unsolicited calls will not be traced back to her.

8.1.2 SPIT vs. SPAM: A Comparison

There are several factors that indicate SPIT might become worse than SPAM. Most have to do with the nature of voice calls as compared to emails.

For example, there is no doubt that a phone call is more intrusive than a new email message. In the first case the phone rings and the user is usually forced to answer or at least check the caller ID, taking her attention away from whatever it is that she is doing, while in the case of an email message she can check her inbox every few hours. In other words, it is easier to ignore a "new mail" warning than the constant ringing of a phone. If SPIT is left uncontrolled to become as common as SPAM, it will indeed lead to situations worse than SPAM could ever

URL: http://www.asterisk.org/
URL: http://www.voipbuster.com
URL: http://www.freecall.com
URL: http://www.voipdiscount.com
be blamed for. An office where each phone rings at the same rate users receive SPAM email messages today is certainly not an office that someone can work in.

Furthermore, it will be even harder to filter out SPIT calls than it is to filter SPAM messages. There are a lot of SPAM filtering devices and applications available in the market and while they do have their weaknesses in that a few unsolicited messages manage to go through and, even worse, legitimate email messages are sometimes erroneously blocked, they can indeed relieve an organization from the problem of SPAM to a large extent. In the case of SPIT mitigation is not as simple. Of course, the Caller ID field cannot be used to differentiate between legitimate and unsolicited calls since it is easily spoofed, so White Lists and Black Lists of phone numbers cannot be considered an 100% effective option, but this is also a problem in the case of SPAM emails and the "From:" header so there is no real difference there. The main difference is that email is not a real-time service. Emails can be stored in a server which gives enough time for them to be analyzed, differentiating SPAM from legitimate messages. This, unfortunately, is not an easy task in the case of a real time service such as the IP Telephony. Taking for granted that there Caller ID cannot be trusted, there is no way a user can find out who is calling her unless she picks up the phone. An incoming SPIT call can, of course, be stored if the user does not pick up the phone and the voicemail system takes over, but this is not any better because having to go through and delete 30 or 40 SPIT messages in order to hear 3-4 legitimate voicemail messages is not practical. One possible solution would be to record calls in the voicemail system, use voice recognition software to generate a text file and then parse that file using the SPAM filters already deployed. The effectiveness of such a system is questionable though and its application is strictly limited to non-real time aspects of the IP telephony service. One can only assume that its false positive and false negative rates will be high since the weaknesses of the voice recognition applications (which are still not really effective - especially if not trained to recognise a specific voice) in their speech-to-text conversion will be added to the weaknesses of the SPAM filters in differentiating between legitimate and SPAM messages.

8.1.3 Generating SPIT

There are only a handful of tools that can be used to effectively generate SPIT calls at the moment but this is not a real obstacle for attackers since someone determined enough can easily develop such an application. All that is needed for basic automated SPIT generation is a simple program (like spitter\textsuperscript{6} that can read call information from a file and create "call files" which should then be moved into the Asterisk outgoing calls folder (path: /var/spool/asterisk/outgoing). As soon as a call file is moved into that folder, the Asterisk PBX will initiate the call. TeleYapper\textsuperscript{7} is a free application that can be abused to provide is a more sophisticated and feature-rich tool for SPIT generation. This application communicates with an SQL database to get usernames, user groups and pre-recorded audio messages and, among other features, it can be configured to call again any phones that did not answer.

8.1.4 Countermeasures

At the moment SPIT is not an important issue for enterprise VoIP deployments since most of these deployments consist of internal networks, disconnected from other enterprise VoIP deployments and the PSTN. In addition, since most companies still use the PSTN to communicate outside their internal network, they will probably not be an attractive SPIT target for early-adopters of this new scheme since VoIP-to-PSTN calls are subject to standard rates in most

\textsuperscript{6}URL: \url{http://www.hackingvoip.com/tools/spitter.tar.gz}

\textsuperscript{7}URL: \url{http://nerdvittles.com/index.php?p=113}
cases. As VoIP becomes widely adopted by organizations and carrier companies, VoIP-to-VoIP calls will become more common, either via the Internet or via VoIP service providers. Call rates will decrease and might even reach a point where VoIP calls will be almost free (e.g. only having to pay a monthly fee that includes unlimited calls). If certain countermeasures are not deployed, SPIT can become a serious issue.

First and foremost Spam over Internet Telephony is a social threat. The drivers behind this threat are not an issue that an enterprise can deal with nor is the adoption of industry-wide countermeasures that will not allow SPIT to become a serious problem. Some of the countermeasures required to mitigate the SPIT threat entail the whole VoIP community working to that end.

Governments can be forced to pass laws that ban SPIT. IP Telephony service users could subscribe to a "do not call" list which contains the numbers of people that do not want to and must not be bothered with unsolicited calls. In the case of telemarketing this does work, which is proved by a similar law that is in effect in the United States where citizens join the National Do Not Call Registry. Telemarketing companies do not have the right to call anyone who has been listed on the Registry for more than 31 days and if they do the callee has the right to file a complaint against the offending company. Arguably, it is not easy to guess the effectiveness of such a countermeasure since, as already mentioned, SPIT is quite different in nature than standard telemarketing calls. SPIT has more in common with SPAM, which is really hard to control and to track people who distribute it (unlike telemarketing and legitimate companies who offer such services).

A really effective countermeasure would be to improve SIP in a way that would make it hard to spoof the caller’s identity. If this becomes a reality then other countermeasures such as black and white lists will become more effective and this will be a big step towards mitigating the SPIT threat. The use of digest authentication combined with the use of TLS between SIP phones and SIP proxies can provide a channel through which users can securely authenticate within their SIP domain. This way, if someone wants to call a user belonging to another domain, the caller’s identity can be assured. Moreover, certain enhancements for SIP’s authenticated identity could be implemented, as defined in RFC 4474 (see Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP), [97]). Quoting the aforementioned IETF document:

> The existing security mechanisms in the Session Initiation Protocol (SIP) are inadequate for cryptographically assuring the identity of the end users that originate SIP requests, especially in an interdomain context. This document defines a mechanism for securely identifying originators of SIP messages. It does so by defining two new SIP header fields, Identity, for conveying a signature used for validating the identity, and Identity-Info, for conveying a reference to the certificate of the signer.

In brief, RFC 4474 describes the use of an enhanced SIP authentication which involves an authentication server (typically the authentication service will be running on the SIP proxy). This server is responsible for authenticating users that try to initiate new calls and then it must compute and sign a hash which will be included in a new header field. This value is computed using the contents of the From: and other fields in the INVITE Request and, consequently, this new value can be used at the destination domain as a means of verifying the identity of the sender. As discussed, it is indeed possible to determine the identity of a caller in a secure manner and

---

8URL: https://www.donotcall.gov
thus build a solid foundation on top of which further countermeasures can be built that will mitigate the SPAM over Internet Telephony threat (and more). Still, if authenticated identity is to be effective, the mechanisms described must be widely deployed and implemented, not only industry-wide but by service providers as well. Unfortunately, it is doubtful if this is going to happen anytime soon since the drivers for such a change and the investments required are not there yet.

When it comes to SPIT countermeasures that an enterprise can implement on its own most solutions resemble the ones already utilized in most enterprises today in order to address the SPAM threat.

Black Lists and White Lists are lists of known attackers and lists of known legitimate users, respectively. If a Black list is used then everyone will be able to make calls to the target organization except for the attackers who are identified on the list. Black lists are unlikely to be effective because, as proven in the case of SPAM, attackers can easily obtain new phone numbers/email addresses or spoof the source of the message/call. On the other hand, in a white list approach only calls originating from numbers in the list will be accepted. This requires a means to allow users to define people whom they trust so that communication with them is allowed. Although white lists are more effective hackers could possibly acquire some phone numbers present in the white list and spoof their calls to pretend they are calling from that number. Moreover, they can block many calls from people who genuinely want to talk with someone in the enterprise but did not have the chance to ask to be included in the list. To deal with that issue approval systems could be used that utilize both black and white lists. In this case, whenever a new user calls, the callee is asked to decide if she wants to accept the call or not. If she does the caller is put on the white list and if not the caller is put on the black list for future reference. It should be noted though that this mechanism adds a significant overhead and could be exploited to flood the users with approval requests.

As described earlier, voicemail messages can be converted to text files and parsed by normal SPAM filters. This can only protect users from SPIT voicemail messages and could have high false positive and false negative rates.

Finally, Voice versions of Completely Automated Public Turing Tests to Tell Computers and Humans Apart (CAPTCHAs) could be used. These tests are simple challenges presented in a way or requiring actions that make sure only humans can answer. CAPTCHAs are widely used in various websites when trying to protect a service (such as a registration or a search engine) from bot-like tools. A widely used implementation is that of a random text message or number embedded in an image with high background noise, such as a surface with lots of dots of various colors, so that humans can discern the text easily while at the same time a computer would have trouble doing so. In order to pass the test and proceed a user must type the correct letter sequence in the field provided. Following the same notion, voice CAPTCHAs are audio messages that a human could easily understand while a computer running voice recognition software would have trouble doing so. This could involve high background noise and a female voice uttering a number sequence asking the user to press the corresponding keys on the handset and then checking the DTMF codes to decide if the user passed the test or not. Voice CAPTCHAs can indeed be very effective but can perhaps irritate legitimate users of the service. A good policy would be to combine CAPTCHAs with black and white lists and only have new users take the test.
In fact, in January 2007, NEC announced\(^9\) an innovative solution called VoIP Seal which helps defend against SPIT by using various techniques including a CAPTCHA. People at NEC claim their technology can identify 99% of SPIT by monitoring communications patterns and blocking SPIT calls before they manage to connect to the user.

A number of vendors, in their latest generation of security appliances, include SPIT mitigation features. Examples include Borderware’s SIPassure\(^{10}\) and Sipera’s IPCS Products\(^{11}\).

A comparison of SPAM and SPIT and a thorough research on the applicability of unsolicited email countermeasures to their voice equivalent can be found in [101].

\(^{9}\)URL: http://www.nec.co.jp/press/en/0701/2602.html
\(^{10}\)URL: http://www.borderware.com/products/sipassure/

89
8.2 VoIP Phishing - Vishing

Phishing is a type of attack where a malicious user or, more often, a group of malicious users utilize social engineering and technical subterfuge to carry out an identity theft and/or a financial account credentials theft. Traditional phishing attacks typically involve email messages which appear to be sent by a legitimate financial organization (e.g. eBay, Paypal, HSBC Bank etc.) and a spoofed website that is usually a copycat version of the organization’s legitimate website. The email messages are sent either in bulk as ordinary SPAM emails or, in more sophisticated attacks, to targeted users that the attackers know are clients of the targeted organization. The goal of the email message is to lure the victim into visiting the spoofed website and disclose sensitive information (e.g. passwords, credit card numbers, PIN numbers, mother maiden’s name etc.) - which will of course be captured by the phishers.

Although the first phishing attacks took place in the 90’s it was not until 2004 that phishing took off and started to be considered a fully developed attack scheme. Since then, it has grown to be a one of the prevalent forms of cyber attacks. According to data from the Anti-Phishing Working Group\(^\text{12}\) (APWG), which is "the global pan-industrial and law enforcement association focused on eliminating the fraud and identity theft that result from phishing, pharming and email spoofing of all types", in January 2004 there were 174 phishing websites reported and by the end of the same year the identified phishing websites were 1,700. APWG’s Phishing Activity Trends Report\(^\text{13}\) for the month of June 2007 mentions that a total of 31,709 unique phishing websites were detected. Figures 8.3 and 8.4, taken from the aforementioned APWG report, show the domination of financial-services-oriented phishing attacks and the top 10 countries that phishing websites are hosted in, respectively.

![Figure 8.3: Most Targeted Industry Sectors in June 2007](http://www.antiphishing.org/report/june_2007.pdf)

\(^{12}\text{URL: http://www.antiphishing.org}\)

\(^{13}\text{URL: http://www.antiphishing.org/reports/apwg_report_june_2007.pdf}\)
Voice Phishing or Vishing, involves an email message which, just like ordinary phishing attacks, is needed to lure the victim into following to the next step of the attack. In the case of Vishing the next step involves an Interactive Voice Response (IVR) system. The IVR is used instead of the spoofed website that traditional phishing attacks usually employ. Victims follow the email’s instructions and call an 800 number that looks legitimate (can be easily bought online) and which has all its calls routed to the malicious IVR. By communicating with the IVR the victims are tricked into saying or entering sensitive information through their handset - the same information that any phishing attack is after (credit card numbers etc.). The attackers record the call and can then play it back to get any information they are after, which includes decoding the DTMF tones.

On June 23, 2006, Websense Security Labs announced the discovery of one of the first vishing attacks deployed. The luring email message was the following:

Subject: Message 156984 Client’s Details Confirmation (Santa Barbara Bank & Trust).

Dear Customer,

We’ve noticed that you experienced trouble logging into Santa Barbara Bank & Trust Online Banking.

After three unsuccessful attempts to access your account, your Santa Barbara Bank & Trust Online Profile has been locked. This has been done to secure your accounts and to protect your private information. Santa Barbara Bank & Trust is committed to make sure that your online transactions are secure.

Call this phone number (1-805-XXX-XXXX) to verify your account and your identity.

Sincerely,
Santa Barbara Bank & Trust Inc.
Online Customer Service

The attack involved a fake IVR and the 1-805 phone number was bought online using a stolen credit card, while, when calling the phone numbers victims were greeted by a pre-recorded message that said: "Welcome to account verification. Please type your 16-digit card number".

Many more attacks have followed and as time goes by these attacks become more and more well planned and executed. One such example is a more recent vishing attack targeted to Bank of America customers and which was reported in the Washington Post on March 2007 (see [8]). It is certain that these are only the first steps of vishing but still they are indicative of things to come. The benefits VoIP can bring to phishers have not gone unnoticed.

It is almost trivial to launch a vishing attack, as was proved by Jay Schulman who, at the 2006 Black Hat Briefings Convention, demonstrated a proof-of-concept vishing attack utilizing only open source tools. Schulman showed that all that a hacker needs to do is buy an 800 number from a VoIP provider, download and install the open source Trixbox\textsuperscript{15} Asterisk-based PBX phone system on a computer and setup Trixbox to use the 800 line and the audio files which the attacker will have recorded from the legitimate IVR system that the target organizations uses and which she is trying to mimic. As soon as the attacker has everything setup, the only thing to do is lure the victims to the malicious IVR. This can be done in a number of ways - which is one of the reasons why Vishing can become a bigger threat than traditional phishing. The attacker can choose to sent emails, as previously described, or she could choose to combine Vishing and SPIT and start calling thousands of random numbers and leaving pre-recorded official-sounding voicemail messages to her victims, informing them of all the disastrous things that can happen them if they do not immediately call at her 800-line. Finally, the attacker could even employ VoIP-to-SMS messages, which is another cost-effective option for IP telephony, so that she bombards various mobile phones with text messages aimed to lure the victims to the malicious IVR. Figure 8.5 demonstrates a vishing attack. More information on the subject of vishing and detailed steps on how to launch a vishing attack can be found in [106].

\textsuperscript{15}URL: http://www.trixbox.org/
Vishing can be a lot more effective than phishing because it relies on the fact that people trust a phone number more than a URL found in an email. This is especially true nowadays since public awareness on the issue of phishing has been significantly raised by reports on the media and elsewhere so the effectiveness of traditional phishing attacks is limited. Combining vishing with SPIT and other techniques that had not been previously used for this task (like text messages) to lure unsuspected victims into the malicious IVR can maximize the effectiveness of the attack.

The most effective way to prevent vishing attacks within an organisation is to prevent the luring email from reaching the employees in the first place. This involves deploying regular SPAM and Phishing filtering technologies. There is a large number of appliances that offer this type of network appliances since SPAM is a big issue. Anti-SPAM devices and applications can already be considered a mature market so there is probably no need to go into more details. In case the attacker decides to use SPIT as a means of luring her victims, then the countermeasures suggested in the SPIT section (authenticated identity, audio content filtering etc.) will probably manage to block the malicious calls and thus will mitigate the threat this variation of a vishing attack poses. Finally, and to protect from cases where a luring SPIT call goes through to its intended recipient, raising the awareness of the employees to the dangers these new technologies can pose will help in preventing them from calling back to the malicious IVR. It is necessary to confirm the number of the financial institution, for example by checking their website, before calling them back. One last countermeasure would be to have a blacklist of known malicious IVRs although its effectiveness is doubtful since the vishers will probably choose to - or have to, if traced - change their 800-line numbers often.
Voice phishing is an emerging threat that is certain to become mainstream soon enough. Just like with traditional phishing, there is a lot of money to be made with vishing and this guarantees that vishing attacks will become more common. Phishers see their income shrink as user awareness is raised on the subject (through training programs, the media, phishing detection integrated in browsers etc.) and they have to resort to extremely sophisticated phishing attacks to accomplish their task. Vishing opens up new opportunities and they are going to take advantage of it. The industry, governments and end-users should be prepared to deal with the new threats that the adoption of VoIP technology will bring.
Chapter 9

Future Developments
& Further Research

As Voice over IP technology becomes widely adopted its benefits become apparent and will, most probably, be the main telephony service of the converged future. This has made researchers turn their heads to VoIP and its implementations and protocols, trying to address its problems, helping the technology mature and better prepare for large-scale deployments.

One of the main VoIP research fields are the VoIP protocols themselves. These protocols are quite new and have not undergone strict testing and/or have inherent vulnerabilities that are not easy to overcome. As recent research ([55]) has shown, an SRTP vulnerability can completely compromise transport-layer security. Moreover, ZRTP, the protocol that was designed as a security enhancement of SRTP, is vulnerable to a man-in-the-middle attack (see [55]) and, additionally, fails to encrypt DTMF tones, as demonstrated in [113].

An other apparent need is that of a universal solution that will be able to protect both media and signaling traffic for IP telephony systems ([75]). Researchers are trying to address these issues and are introducing various mechanisms such as an authentication and integrity scheme for both audio and media that is based on digital watermarking as presented in [84] and a similar mechanism for PSTN-VoIP hybrid networks presented in [125] but research is still going on in the field.

Moreover, research must be conducted on securing the existing protocols and increasing their robustness. An example of such research is the work presented in [45], which provides a framework to protect SIP-based infrastructures from malformed message attacks.

More research is also needed on the subject of VoIP Quality of Service. Researchers are already working on providing mechanisms that will improve VoIP reliability. [70] presents an RTP/RTPC-based method used to detect network problems and, using overlay networks, reroute traffic via healthy network segments.

Wireless networks are another field where VoIP will probably become widely adopted but it seems the technology is not able to support large-scale wireless deployments yet. There are already numerous large-scale wireless network deployments implemented and perhaps the day when wireless Internet connections will be available throughout big cities is not that far. In such a scenario, wireless VoIP networks and handsets could potentially replace mobile phones, which would bring significant cost saving benefits to both users and carriers. Unfortunately, VoIP was not built around mobility and there is a lot of work that needs to be done in this area if WiFi and VoIP technologies are to benefit from the drivers behind the adoption of each other. An
example of research in this area is [127], which presents a framework aimed towards creating a mobility enabled wireless IP infrastructure. The unstable connection quality of wireless and is scarce bandwidth are also issues that should be addressed as well as how VoIP QoS solutions can be supported in WiFi networks. [121] tries to address some of the aforementioned issues, proposing several schemes in various protocol layers that could help wireless networks better accommodate the VoIP service. Finally, [118] introduces a new adaptive architecture to address the issue of transporting VoIP traffic over heterogeneous wired/wireless Internet environments.

Finally, further research could also focus on improving VoIP security appliances such as VoIP-aware Firewalls, Intrusion Prevention and Intrusion Detection Systems (VIPS and VIDS), trying to find patterns on the traffic generated by known VoIP attacks to improve the effectiveness of VIPS/VIDS and even focus on improving the effectiveness of anti-SPIT appliances.
Chapter 10

Conclusion

Voice over Internet Protocol brings numerous benefits to businesses, carrier network companies and end-users alike. It should be obvious at this point that the convergence of voice, data and video over packet-switched networks will definitely happen since the benefits from this move are tremendous for almost everyone. Sooner or later, IP networks will become the main channel of telephony - and video - communications.

Nonetheless, VoIP should not be hastily adopted in the enterprise sector. Deploying VoIP can introduce a number of threats to the confidentiality, integrity and availability of sensitive corporate information and services. It must be made clear that having a secure VoIP is not an easy goal to accomplish. This is because of the inherent security problems the IP telephony service has. Since voice is run over IP, it is exposed to all the security weaknesses of the IP protocol (sniffing, replay attacks, spoofing etc.) and it is also exposed to the vulnerabilities of the supporting protocols (e.g. DNS) and the VoIP protocols themselves (SIP, H.323, MGCP, RPT etc.). Then there is always the issue of physical security. Moreover, the IP telephony service depends on the security of the underlying infrastructure, be it voice (phones, SIP proxys etc.) or data (routers, switches, firewalls etc.). It should also be reminded that, unlike the PSTN, VoIP signaling and media traffic share the same network which opens up many opportunities for misuse. The different VoIP architectures and protocols, the variations in speech coding techniques and the various combinations of those only make things worse when trying to analyze and respond to the different security risks a VoIP deployment might be exposed to. Furthermore, managing access to the VoIP network, managing traffic priority and last but not least ensuring network availability introduce significant network management overhead. Finally, the real-time nature of voice communications and the extreme sensitivity of the human ear to slight sound quality variations introduce new variables in the VoIP security equation (like jitter, latency and packet loss) and only exacerbate the aforementioned issues.

Figure 10.1 shows a detailed list of the threats a VoIP deployment can be exposed to, categorized by network layers.
<table>
<thead>
<tr>
<th>Vulnerability</th>
<th>Confidentiality</th>
<th>Integrity</th>
<th>Availability</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data Link</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Physical Attacks</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>ARP cache</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>ARP flood</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAC spoofing</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td><strong>Internet</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IP spoofing</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Registration server, IP phone, MGCP, DNS, etc</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Redirect via IP spoof</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Malformed packets</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>IP frag</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Jolt</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Transport</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TCP / UDP flood</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TCP / UDP replay</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Application</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TFTP server insertion</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DHCP server insertion (redirect)</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>DHCP IP address starvation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICMP flood</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SIP</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Registration Hijacking</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Call Hijacking (MGCP NotifiedEntity parameter)</td>
<td></td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Message body modification</td>
<td>x</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>RTP insertion</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Spoof via header</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Cancel / bye attack</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Malformed method</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Redirect method</td>
<td></td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td><strong>RTP</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SDP redirect</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>RTP payload</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RTP message tampering</td>
<td></td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Encryption</td>
<td></td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Default settings / passwords</td>
<td></td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Disable unnecessary services HTTP, FTP, etc</td>
<td></td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Buffer overflow</td>
<td></td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Legacy Network Interaction</td>
<td></td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>DNS Availability</td>
<td></td>
<td>x</td>
<td>x</td>
</tr>
</tbody>
</table>

Figure 10.1: Threats to the Confidentiality, Integrity and Availability of VoIP. ([85])
The above facts about VoIP should not give the impression that a secure VoIP deployment is not feasible. As described throughout this Thesis there are various countermeasures that can be applied to mitigate the risks that the inherent vulnerabilities of IP telephony pose. Logically separating voice and data by using VLANs making sure that DHCP and TFTP servers are separated from the data network as well, using strong authentication and access control on core network management components like Media Gateways, using VoIP-aware Firewalls, IPSs and IDSs, using IPSec, avoiding softphones, hardening platforms that VoIP relies on (disabling services, patching etc.), patching VoIP hardware and keeping it up-to-date with firmware updates and giving special consideration to emergency calls are just some of the countermeasures already suggested which are necessary building blocks in construction a secure VoIP deployment.

To summarize, a suitable security policy must be developed and followed for the whole network and a defense-in-depth layered approach to security should be taken. Most importantly the effectiveness of the safeguards put in place must be continuously monitored and re-evaluated.

Figure 10.2, taken from the *IP Telephony & VoIP Security Technical Implementation Guide* of the U.S. Department of Defense, demonstrates a safe blueprint for VoIP deployments.
Figure 10.2: Safe Blueprint for Voice Over IP

- Data Environment (Physical Network)
- Local Data Network Addressing (Physical Infrastructure)
- Data Network (PLMN), Soft Phone
- Site Enclave

- TDM Network (DSN/PSN)
- VoIP Virtual Network (VLAN, Private Addressing, Routed Data Infrastructure)
- VoIP Switch

- VoIP Servers
- Voice (TDM)
- VoIP Protocol (SIP, RTP)
- Gateway, Mail, SMSC, etc.
It is important to point out that while the countermeasures are available this does not mean that an enterprise can just deploy this new technology, put a countermeasure here and there and wait to reap the benefits of VoIP. It is essential to understand the many aspects of this technology because only then can someone make an informed decision on if it is worth it move to VoIP or not.

VoIP security is a complex problem. Some of the security measures like encryption can add too much processing overhead and affect the QoS. It might prove necessary to install new VoIP aware network devices and that, in order to maintain an acceptable QoS while using encryption, the organization must purchase newer firewall models that feature an encryption processor. All of the above mean extra costs. In order provide to the VoIP deployment the redundancy required for a telecommunications service extra power backup systems may be needed, which is again, extra costs. The deployment of VoIP might even expose the organization to regulatory compliance (e.g. ZRTP) and other threats.

All of the above are not meant to illustrate a dark future for VoIP but voice communications are vital and any changes to the way they are delivered should not be taken lightly. Fortunately, things improve rapidly for Voice over IP as security is being built into its protocols and more and more vendors launch VoIP-oriented and VoIP-aware products with improved effectiveness and reduced prices.

All of the above are not meant to present the deployment of VoIP as an uncertain investment either. The move to Voice over IP is not a technology decision; it is a business decision. Each organization is a unique case and higher management must perform a risk analysis to identify the risks associated with the move to IP telephony and then proceed to map those risks to the organization’s businesses processes, in order to properly quantify those risks. Only then, and taking the organization’s risk appetite into account, will it be possible to make an informed decision on whether the cost savings and business benefits introduced by the move to Voice over IP can outweigh the cost of implementing and securing this new and promising technology.
Chapter 11

Bibliography
Bibliography


[74] Paul D. Kretkowski. Your VoIP Calls are Wiretapped - and Other Legal Surprises, April 2007.

[76] Sipera VIPER Lab. VIPER-2007-010-G Threat Advisory. Some SIP implementations on endpoints may be vulnerable to CPU and battery exhaustion due multiple incompletely setup sessions.


