Chapter 5

Threats to VoIP Communications Systems

Solutions in this chapter:

- **Denial-of-Service or VoIP Service Disruption**
- **Call Hijacking and Interception**
- **H.323-Specific Attacks**
- **SIP-Specific Attacks**

Introduction

Converging voice and data on the same wire, regardless of the protocols used, ups the ante for network security engineers and managers. One consequence of this convergence is that in the event of a major network attack, the organization's entire telecommunications infrastructure can be at risk. Securing the whole VoIP infrastructure requires planning, analysis, and detailed knowledge about the specifics of the implementation you choose to use.

Table 5.1 describes the general levels that can be attacked in a VoIP infrastructure.

Table 5.1 VoIP Vulnerabilities

Vulnerability	Description
IP infrastructure	Vulnerabilities on related non-VoIP systems can lead to compromise of VoIP infrastructure.
Underlying operating system	VoIP devices inherit the same vulnerabilities as the operating system or firmware they run on. Operating systems are Windows and Linux.
Configuration	In their default configuration most VoIP devices ship with a surfeit of open services. The default services running on the open ports may be vulnerable to DoS attacks, buffer overflows, or authentication bypass.
Application level	Immature technologies can be attacked to disrupt or manipulate service. Legacy applications (DNS, for example) have known problems.

Denial-of-Service or VoIP Service Disruption

Denial-of-service (DoS) attacks can affect any IP-based network service. The impact of a DoS attack can range from mild service degradation to complete loss of service. There are several classes of DoS attacks. One type of attack in which packets can simply be flooded into or at the target network from multiple external sources is called a distributed denial-of-service (DDoS) attack (see Figures 5.1 and 5.2).

In this figure, traffic flows normally between internal and external hosts and servers. In Figure 5.2, a network of computers (e.g., a botnet) directs IP traffic at the interface of the firewall.

Figure 5.1 Typical Internet Access

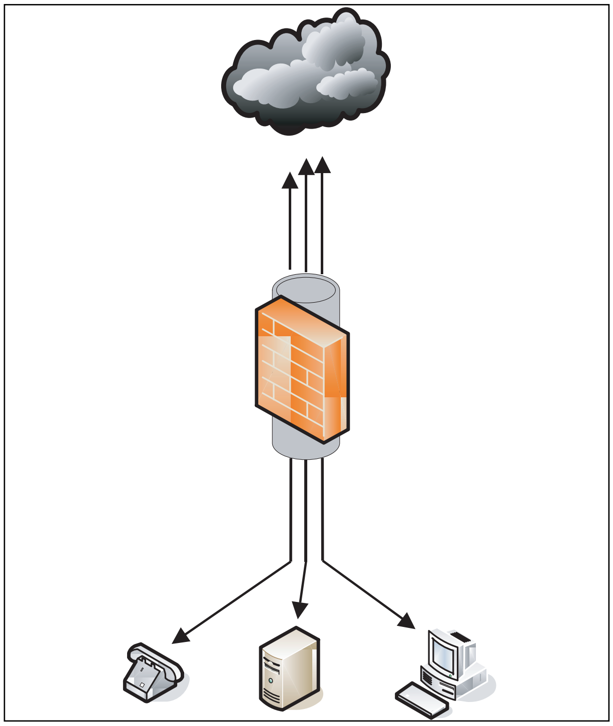
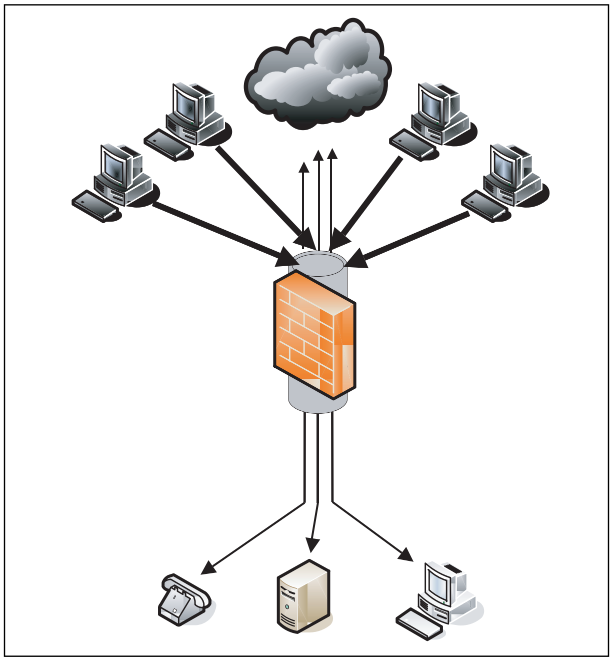
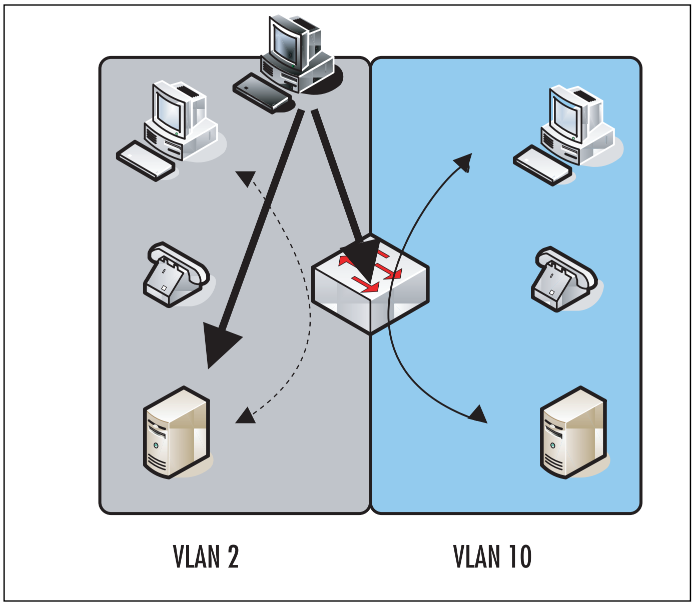


Figure 5.2 A Distributed Denial-of-Service Attack



The second large class of Denial of Service (DoS) conditions occurs when devices within the internal network are targeted by a flood of packets so that they fail—taking out related parts of the infrastructure with them. As in the DdoS scenarios described earlier in this chapter, service disruption occurs to resource depletion—primarily bandwidth and CPU resource starvation (see Figure 5.3). For example, some IP telephones will stop working if they receive a UDP packet larger than 65534 bytes on port 5060.

Figure 5.3 An Internal Denial-of-Service Attack



Neither integrity checks nor encryption can prevent these attacks. DoS or DDoS attacks are characterized simply by the volume of packets sent toward the victim computer; whether those packets are signed by a server, contain real or spoofed source IP addresses, or are encrypted with a fictitious key—none of these are relevant to the attack.

DoS attacks are difficult to defend against, and because VoIP is just another IP network service, it is just as susceptible to DoS attack as any other IP network services. Additionally, DoS attacks are particularly effective against services such as VoIP and other real-time services, because these services are most sensitive to adverse network status. Viruses and worms are included in this category as they often cause DoS or DDoS due to the increased network traffic that they generate as part of their efforts to replicate and propagate.

How do we defend against these DoS conditions (we won't use the term attack here because some DoS conditions are simply the unintended result of other unrelated actions)? Let's begin with internal DoS. Note in Figure 5.3 that VLAN 10 on the right is not affected by the service disruption on the left in VLAN 2. This illustrates one critical weapon the security administrator has in thwarting DoS conditions—logical segregation of network domains in separate compartments. Each compartment can be configured to be relatively immune to the results of DoS in the others. This is described in more detail in Chapter 8.

Point solutions will also be effective in limiting the consequences of DoS conditions. For example, because strong authentication is seldom used in VoIP environments, the message processing components must trust and process messages from possible attackers. The additional processing of bogus messages exhausts server resources and leads to a DoS. SIP or H.323 Registration Flooding is an example of this, described in the list of DoS threats, later. In that case, message processing servers can mitigate this specific threat by limiting the number of registrations it will accept per minute for a particular address (and/or from a specific IP address). An intrusion prevention system (IPS) may be useful in fending off certain types of DoS attacks. These devices sit on the datapath and monitor passing traffic. When anomalous traffic is detected (either by matching against a database of attack signatures or by matching the results of an anomaly-detection algorithm) the IPS blocks the suspicious traffic. One problem I have seen with these devices—particularly in environments with high availability requirements—is that they sometimes block normal traffic, thus creating their own type of DoS.

Additionally, security administrators can minimize the chances of DoS by ensuring that IP telephones and servers are updated to the latest stable version and release. Typically, when a DoS warning is announced by bugtraq, the vendor quickly responds by fixing the offending software.

NOTE

VoIP endpoints can be infected with new VoIP device or protocol-specific viruses. WinCE, PalmOS, SymbianOS, and POSIX-based softphones are especially vulnerable because they typically do not run antivirus software and have less robust operating systems. Several Symbian worms already have been detected in the wild. Infected VoIP devices then create a new "weak link" vector for attacking other network resources.

Compromised devices can be used to launch attacks against other systems in the same network, particularly if the compromised device is trusted (i.e., inside the firewall). Malicious programs installed by an attacker on compromised devices can capture user input, capture traffic, and relay user data over a "back channel" to the attacker. This is especially worrisome for softphone users.

VoIP systems must meet stringent service availability requirements. Following are some example DoS threats can cause the VoIP service to be partially or entirely unavailable by preventing successful call placement (including emergency/911), disconnecting existing calls, or preventing use of related services like voicemail. Note that this list is not exhaustive but illustrates some attack scenarios.

- **TLS Connection Reset** It's not hard to force a connection reset on a TLS connection (often used for signaling security between phones and gateways)—just send the right kind of junk packet and the TLS connection will be reset, interrupting the signaling channel between the phone and call server.
- **VoIP Packet Replay Attack** Capture and resend out-of-sequence VoIP packets (e.g., RTP SSRC—SSRC is an RTP header field that stands for Synchronization Source) to endpoints, adding delay to call in progress and degrading call quality.
- **Data Tunneling** Not exactly an attack; rather tunneling data through voice calls creates, essentially, a new form of unauthorized modem. By transporting modem signals through a packet network by using pulse code modulation (PCM) encoded packets or by residing within header information, VoIP can be used to support a modem call over an IP network. This technique may be used to bypass or undermine a desktop modem policy and hide the existence of unauthorized data connections. This is similar in concept to the so-called "IP over HTTP" threat (i.e., "Firewall Enhancement Protocol" RFC 3093)—a classic problem for any ports opened on a firewall from internal sources.
- **QoS Modification Attack** Modify non-VoIP-specific protocol control information fields in VoIP data packets to and from endpoints to degrade or deny voice service. For example, if an attacker were to change 802.1Q VLAN tag or IP packet ToS bits, either as a man-in-the-middle or by compromising endpoint device configuration, the attacker could disrupt the quality of service "engineered" for a VoIP network. By subordinating voice traffic to data traffic, for example, the attacker might substantially delay delivery of voice packets.
- **VoIP Packet Injection** Send forged VoIP packets to endpoints, injecting speech or noise or gaps into active call. For example, when RTP is used without authentication of RTCP packets (and without SSRC sampling), an attacker can inject RTCP packets into a multicast group, each with a different SSRC, which can grow the group size exponentially.
- **DoS against Supplementary Services** Initiate a DoS attack against other network services upon which the VoIP service depends (e.g., DHCP, DNS, BOOTP). For example, in networks where VoIP endpoints rely on DHCP-assigned addresses, disabling the DHCP server prevents endpoints (soft- and hardphones) from

acquiring addressing and routing information they need to make use of the VoIP service.

- **Control Packet Flood** Flood VoIP servers or endpoints with unauthenticated call control packets, (e.g., H.323 GRQ, RRQ, URQ packets sent to UDP/1719). The attacker's intent is to deplete/exhaust device, system, or network resources to the extent that VoIP service is unusable. Any open administrative and maintenance port on call processing and VoIP-related servers can be a target for this DoS attack.
- **Wireless DoS** Initiate a DoS attack against wireless VoIP endpoints by sending 802.11 or 802.1X frames that cause network disconnection (e.g., 802.11 Deauthenticate flood, 802.1X EAP-Failure, WPA MIC attack, radio spectrum jamming). For example, a Message Integrity Code attack exploits a standard countermeasure whereby a wireless access point disassociates stations when it receives two invalid frames within 60 seconds, causing loss of network connectivity for 60 seconds. In a VoIP environment, a 60-second service interruption is rather extreme.
- **Bogus Message DoS** Send VoIP servers or endpoints valid-but-forged VoIP protocol packets to cause call disconnection or busy condition (e.g., RTP SSRC collision, forged RTCP BYE, forged CCMS, spoofed endpoint button push). Such attacks cause the phone to process a bogus message and incorrectly terminate a call, or mislead a calling party into believing the called party's line is busy.
- **Invalid Packet DoS** Send VoIP servers or endpoints invalid packets that exploit device OS and TCP/IP implementation denial-of-service CVEs. For example, the exploit described in CAN-2002-0880 crashes Cisco IP phones using jolt, jolt2, and other common fragmentation-based DoS attack methods. CAN-2002-0835 crashes certain VoIP phones by exploiting DHCP DoS CVEs. Avaya IP phones may be vulnerable to port zero attacks.
- **Immature Software DoS** PDA/handheld softphones and first generation VoIP hardphones are especially vulnerable because they are not as mature or intensely scrutinized. VoIP call servers and IP PBXs also run on OS platforms with many known CVEs. Any open administrative/maintenance port (e.g., HTTP, SNMP, Telnet) or vulnerable interface (e.g., XML, Java) can become an attack vector.
- **VoIP Protocol Implementation DoS** Send VoIP servers or endpoints invalid packets to exploit a VoIP protocol implementation vulnerability to a DoS attack. Several such exploits are identified in the MITRE CVE database (http://cve.mitre.org). For example, CVE-2001-00546 uses malformed H.323 packets to exploit Windows ISA memory leak and exhaust resources. CAN-2004-0056 uses malformed H.323 packets to exploit Nortel BCM DoS vulnerabilities. Lax software update practices (failure to install CVE patches) exacerbate risk.

- **Packet of Death DoS** Flood VoIP servers or endpoints with random TCP, UDP, or ICMP packets or fragments to exhaust device CPU, bandwidth, TCP sessions, and so on. For example, an attacker can initiate a TCP Out of Band DoS attack by sending a large volume of TCP packets marked "priority delivery" (the TCP Urgent flag). During any flood, increased processing load interferes with the receiving system's ability to process real traffic, initially delaying voice traffic processing but ultimately disrupting service entirely.
- **IP Phone Flood DoS** Send a very large volume of call data toward a single VoIP endpoint to exhaust that device's CPU, bandwidth, TCP sessions, and so on. Interactive voice response systems, telephony gateways, conferencing servers, and voicemail systems are able to generate more call data than a single endpoint can handle and so could be leveraged to flood an endpoint.

Call Hijacking and Interception

Call interception and eavesdropping are other major concerns on VoIP networks. The VOIPSA threat taxonomy (www.voipsa.org/Activities/taxonomy-wiki.php) defines eavesdropping as "a method by which an attacker is able to monitor the entire signaling and/or data stream between two or more VoIP endpoints, but cannot or does not alter the data itself." Successful call interception is akin to wiretapping in that conversations of others can be stolen, recorded, and replayed without their knowledge. Obviously, an attacker who can intercept and store these data can make use of the data in other ways as well.

Tools & Traps…

DNS Poisoning

A DNS A (or address) record is used for storing a domain or hostname mapping to an IP address. SIP makes extensive use of SRV records to locate SIP services such as SIP proxies and registrars. SRV (service) records normally begin with an underscore (_sip.tcpserver.udp.domain.com) and consist of information describing service, transport, host, and other information. SRV records allow administrators to use several servers for a single domain, to move services from host to host with little fuss, and to designate some hosts as primary servers for a service and others as backups.

An attacker's goal, when attempting a DNS Poisoning or spoofing attack, is to replace valid cached DNS A, SRV, or NS records with records that point to the attacker's server(s). This can be accomplished in a number of fairly trivial ways—the easiest being to initiate a zone transfer from the attacker's DNS server to the victim's misconfigured

Continued

DNS server, by asking the victim's DNS server to resolve a networked device within the attacker's domain. The victim's DNS server accepts not only the requested record from the attacker's server, but it also accepts and caches any other records that the attacker's server includes.

Thus, in addition to the A record for www.attacker.com, the victim DNS server may receive a bogus record for www.yourbank.com. The innocent victim will then be redirected to the attacker.com Web site anytime he or she attempts to browse to the yourbank.com Web site, as long as the bogus records are cached. Substitute a SIP URL for a Web site address, and the same scenario can be repeated in a VoIP environment.

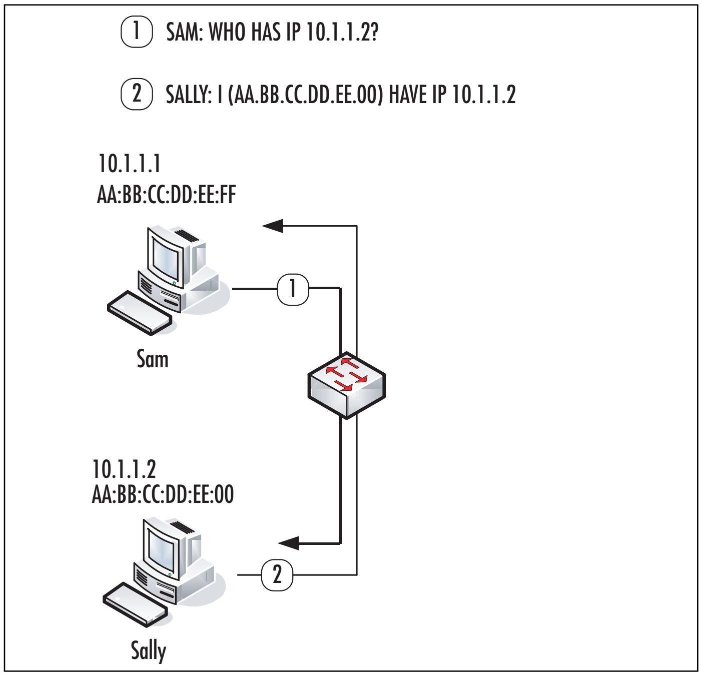
This family of threats relies on the absence of cryptographic assurance of a request's originator. Attacks in this category seek to compromise the message integrity of a conversation. This threat demonstrates the need for security services that enable entities to authenticate the originators of requests and to verify that the contents of the message and control streams have not been altered in transit.

In the past several years, as host PCs have improved their processing power and their ability to process networked information, network administrators have instituted a hierarchical access structure that consists of a single, dedicated switched link for each host PC to distribution or backbone devices. Each networked user benefits from a more reliable, secure connection with guaranteed bandwidth. The use of a switched infrastructure limits the effectiveness of packet capture tools or protocol analyzers as a means to collect VoIP traffic streams. Networks that are switched to the desktop allow normal users' computers to monitor only broadcast and unicast traffic that is destined to their particular MAC address. A user's NIC (network interface card) literally does not see unicast traffic destined for other computers on the network.

The address resolution protocol (ARP) is a method used on IPv4 Ethernet networks to map the IP address (layer 3) to the hardware or MAC (Media Access Control) layer 2 address. (Note that ARP has been replaced in IPv6 by Neighbor Discovery [ND] protocol. The ND protocol is a hybrid of ARP and ICMP.) Two classes of hardware addresses exist: the broadcast address of all ones, and a unique 6 byte identifier that is burned into the PROM of every NIC (Network Interface Card).

Figure 5.4 illustrates a typical ARP address resolution scheme. A host PC (10.1.1.1) that wishes to contact another host (10.1.1.2) on the same subnet issues an ARP broadcast packet (ARPs for the host) containing its own hardware and IP addresses. NICs contain filters that allow them to drop all packets not destined for their unique hardware address or the broadcast address, so all NICs but the query target silently discard the ARP broadcast. The target NIC responds to the query request by unicasting its IP and hardware address, completing the physical to logical mapping, and allowing communications to proceed at layer 3.

Figure 5.4 Typical ARP Request/Reply


1 SAM: WHO HAS IP 10.1.1.2?
2 SALLY: I (AA.BB.CC.DD.EE.00) HAVE IP 10.1.1.2
10.1.1.1
AA:BB:CC:DD:EE:FF
1
Sam
10.1.1.2
AA:BB:CC:DD:EE:00
2
Sally

To minimize broadcast traffic, many devices cache ARP addresses for a varying amount of time: The default ARP cache timeout for Linux is one minute; for Windows NT, two minutes, and for Cisco routers, four hours. This value can be trivially modified in most systems. The ARP cache is a table structure that contains IP address, hardware address, and oftentimes, the name of the interface the MAC address is discovered on, the type of media, and the type of ARP response. Depending upon the operating system, the ARP cache may or may not contain an entry for its own addresses.

In Figure 5.4, Sam's ARP cache contains one entry prior to the ARP request/response:

Internet Address	Physical Address	
10.1.1.1	AA:BB:CC:DD:EE:FF	int0

After the ARP request/response completes, Sam's ARP cache now contains two entries:

Internet Address	Physical Address	
10.1.1.1	AA:BB:CC:DD:EE:FF	int0
10.1.1.2	AA:BB:CC:DD:EE:00	int0

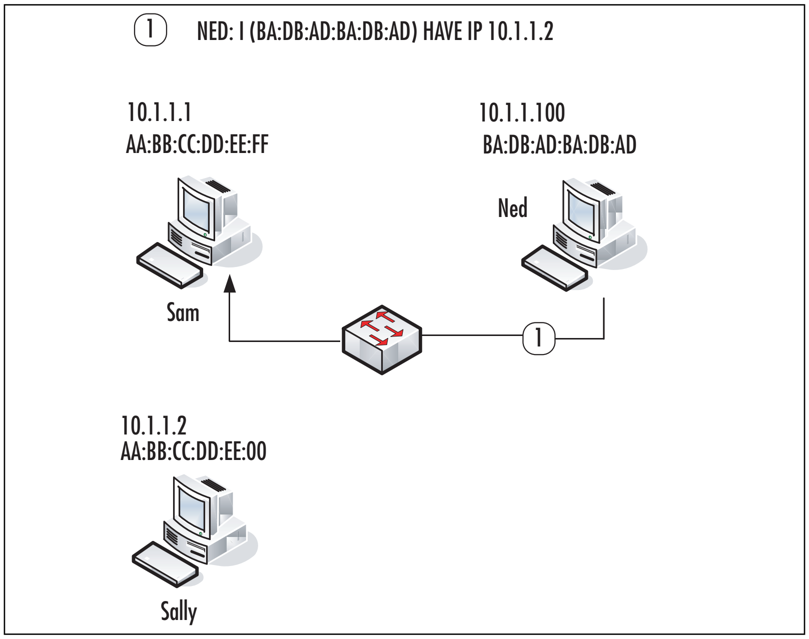
Note that Sally's ARP cache, as a result of the request/response communications, is updated with the hardware:IP mappings for both workstations as well.

ARP Spoofing

ARP is a fundamental Ethernet protocol. Perhaps for this reason, manipulation of ARP packets is a potent and frequent attack mechanism on VoIP networks. Most network administrators assume that deploying a fully switched network to the desktop prevents the ability of network users to sniff network traffic and potentially capture sensitive information traversing the network. Unfortunately, several techniques and tools exist that allow any user to sniff traffic on a switched network because ARP has no provision for authenticating queries or query replies. Additionally, because ARP is a stateless protocol, most operating systems (Solaris is an exception) update their cache when receiving ARP reply, regardless of whether they have sent out an actual request.

Among these techniques, ARP redirection, ARP spoofing, ARP hijacking, and ARP cache poisoning are related methods for disrupting the normal ARP process. These terms frequently are interchanged and confused. For the purpose of this section, we'll refer to ARP cache poisoning and ARP spoofing as the same process. Using freely available tools such as ettercap, Cain, and dsniff, an evil IP device can spoof a normal IP device by sending unsolicited ARP replies to a target host. The bogus ARP reply contains the hardware address of the normal device and the IP address of the malicious device. This "poisons" the host's ARP cache (see Figure 5.5).

Figure 5.5 ARP Spoofing (Cache Poisoning)



In Figure 5.5, Ned is the attacking computer. When SAM broadcasts an ARP query for Sally's IP address, Ned, the attacker, responds to the query stating that the IP address

(10.1.1.2) belongs to Ned's MAC address, BA:DB:AD:BA:DB:AD. Packets sent from Sam supposedly to Sally will be sent to Ned instead. Sam will mistakenly assume that Ned's MAC address corresponds to Sally's IP address and will direct all traffic destined for that IP address to Ned's MAC. In fact, Ned can poison Sam's ARP cache without waiting for an ARP query since on Windows systems (9x/NT/2K), static ARP entries are overwritten whenever a query response is received regardless of whether or not a query was issued.

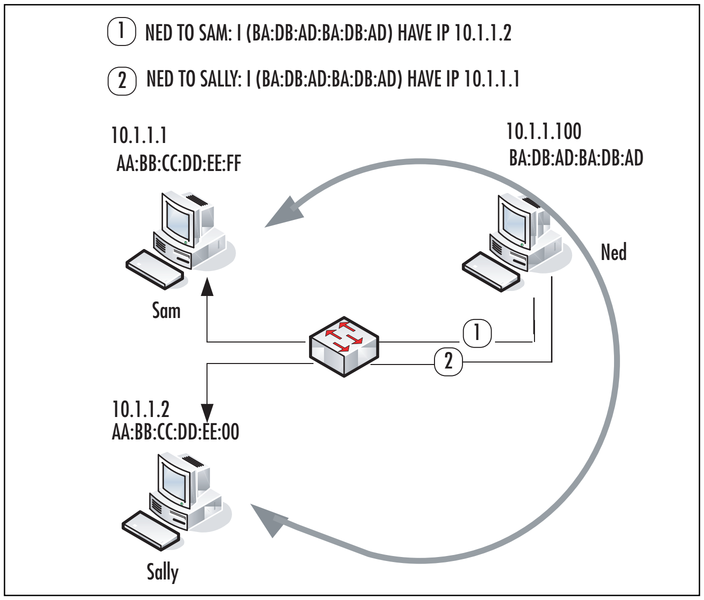
Sam's ARP cache now looks like this:

Internet Address	Physical Address	
10.1.1.1	AA:BB:CC:DD:EE:FF	int0
10.1.1.2	BA:DB:AD:BA:DB:AD	int0

This entry will remain until it ages out or a new entry replaces it.

ARP redirection can work bidirectionally, and a spoofing device can insert itself in the middle of a conversation between two IP devices on a switched network (see Figure 5.6). This is probably the most insidious ARP-related attack. By routing packets on to the devices that should truly be receiving the packets, this insertion (known as a Man/Monkey/Moron in the Middle attack) can remain undetected for some time. An attacker can route packets to /dev/null (nowhere) as well, resulting in a DoS attack.

Figure 5.6 An ARP MITM Attack


1 NED TO SAM: I (BA:DB:AD:BA:DB:AD) HAVE IP 10.1.1.2
2 NED TO SALLY: I (BA:DB:AD:BA:DB:AD) HAVE IP 10.1.1.1
10.1.1.1
AA:BB:CC:DD:EE:FF
10.1.1.100
BA:DB:AD:BA:DB:AD
Ned
Sam
1
2
10.1.1.2
AA:BB:CC:DD:EE:00
Sally

Sam's ARP cache:

Internet Address	Physical Address	
10.1.1.1	AA:BB:CC:DD:EE:FF	int0
10.1.1.2	BA:DB:AD:BA:DB:AD	int0

Sally's ARP cache:

Internet Address	Physical Address	
10.1.1.1	BA:DB:AD:BA:DB:AD	int0
10.1.1.2	AA:BB:CC:DD:EE:00	int0

As all IP traffic between the true sender and receiver now passes through the attacker's device, it is trivial for the attacker to sniff that traffic using freely available tools such as Ethereal or tcpdump. Any unencrypted information (including e-mails, usernames and passwords, and web traffic) can be intercepted and viewed.

This interception has potentially drastic implications for VoIP traffic. Freely available tools such as vomit and rtpsniff, as well as private tools such as VoipCrack, allow for the interception and decoding of VoIP traffic. Captured content can include speech, signaling and billing information, multimedia, and PIN numbers. Voice conversations traversing the internal IP network can be intercepted and recorded using this technique.

There are a number of variations of the aforementioned techniques. Instead of imitating a host, the attacker can emulate a gateway. This enables the attacker to intercept numerous packet streams. However, most ARP redirection techniques rely on stealth. The attacker in these scenarios hopes to remain undetected by the users being impersonated. Posing as a gateway may result in alerting users to the attacker's presence due to unanticipated glitches in the network, because frequently switches behave in unexpected ways when attackers manipulate ARP processes. One unintended (much of the time) consequence of these attacks, particularly when switches are heavily loaded, is that the switch CAM (Content-Addressable Memory) table—a finite-sized IP address to MAC address lookup table—becomes disrupted. This leads to the switch forwarding unicast packets out many ports in unpredictable fashion. Penetration testers may want to keep this in mind when using these techniques on production networks.

In order to limit damage due to ARP manipulation, administrators should implement software tools that monitor MAC to IP address mappings. The freeware tool, Arpwatch, monitors these pairings. At the network level, MAC/IP address mappings can be statically coded on the switch; however, this is often administratively untenable. Dynamic ARP Inspection (DAI) is available on newer Cisco Catalyst 6500 switches. DAI is part of Cisco's Integrated Security (CIS) functionality and is designed to prevent several layer two and layer three spoofing attacks, including ARP redirection attacks. Note that DAI and CIS are available only on Catalyst switches using native mode (Cisco IOS).

The potential risks of decoding intercepted VoIP traffic can be eliminated by implementing encryption. Avaya's Media Encryption feature is an example of this. Using Media Encryption, VoIP conversations between two IP endpoints are encrypted using AES encryption. In highly secure environments, organizations should ensure that Media Encryption is enabled on all IP codec sets in use.

DAI enforces authorized MAC-to-IP address mappings. Media Encryption renders traffic, even if intercepted, unintelligible to an attacker.

The following are some additional examples of call or signal interception and hijacking. This class of threats, though typically more difficult to accomplish than DoS, can result in significant loss or alteration of data. DoS attacks, whether caused by active methods or inadvertently, although important in terms of quality of service, are more often than not irritating to users and administrators. Interception and hijacking attacks, on the other hand, are almost always active attacks with theft of service, information, or money as the goal. Note that this list is not exhaustive but illustrates some attack scenarios.

- **Rogue VoIP Endpoint Attack** Rogue IP endpoint contacts VoIP server by leveraging stolen or guessed identities, credentials, and network access. For example, a rogue endpoint can use an unprotected wall jack and auto-registration of VOIP phones to get onto the network. RAS password guessing can be used to masquerade as a legitimate endpoint. Lax account maintenance (expired user accounts left active) increases risk of exploitation.
- **Registration Hijacking** Registration hijacking occurs when an attacker impersonates a valid UA to a registrar and replaces the registration with its own address. This attack causes all incoming calls to be sent to the attacker.
- **Proxy Impersonation** Proxy impersonation occurs when an attacker tricks a SIP UA or proxy into communicating with a rogue proxy. If an attacker successfully impersonates a proxy, he or she has access to all SIP messages.
- **Toll Fraud** Rogue or legitimate VoIP endpoint uses a VoIP server to place unauthorized toll calls over the PSTN. For example, inadequate access controls can let rogue devices place toll calls by sending VoIP requests to call processing applications. VoIP servers can be hacked into in order to make free calls to outside destinations. Social engineering can be used to obtain outside line prefixes.
- **Message Tampering** Capture, modify, and relay unauthenticated VoIP packets to/from endpoints. For example, a rogue 802.11 AP can exchange frames sent or received by wireless endpoints if no payload integrity check (e.g., WPA MIC, SRTP) is used. Alternatively, these attacks can occur through registration hijacking, proxy impersonation, or an attack on any component trusted to process SIP or

H.323 messages, such as the proxy, registration servers, media gateways, or firewalls. These represent non-ARP-based MITM attacks.

- **VoIP Protocol Implementation Attacks** Send VoIP servers or endpoints invalid packets to exploit VoIP protocol implementation CVEs. Such attacks can lead to escalation of privileges, installation and operation of malicious programs, and system compromise. For example, CAN-2004-0054 exploits Cisco IOS H.323 implementation CVEs to execute arbitrary code. CSCed33037 uses unsecured IBM Director agent ports to gain administrative control over IBM servers running Cisco VoIP products.

Notes from the Underground…

ANI/Caller-ID Spoofing

Caller ID is a service provided by most telephone companies (for a monthly cost) that will tell you the name and number of an incoming call. Automatic Number Identification (ANI) is a system used by the telephone company to determine the number of the calling party. To spoof Caller-ID, an attacker sends modem tones over a POTS lines between rings 1 and 2. ANI spoofing is setting the ANI so as to send incorrect ANI information to the PSTN so that the resulting Caller-ID is misleading. Traditionally this has been a complicated process either requiring the assistance of a cooperative phone company operator or an expensive company PBX system.

In ANI/Caller-ID spoofing, an evildoer hijacks phone number and the identity of a trusted party, such as a bank or a government office. The identity appears on the caller ID box of an unsuspecting victim, with the caller hoping to co-opt valuable information, such as account numbers, or otherwise engage in malicious mischief. This is not a VoIP issue, per se. In fact, one of the big drawbacks about VoIP trunks is their inability to send ANI properly because of incomplete standards.

H.323-Specific Attacks

The only existing vulnerabilities that we are aware of at this time take advantage of ASN.1 parsing defects in the first phase of H.225 data exchange. More vulnerabilities can be expected for several reasons: the large number of differing vendor implementations, the complex nature of this collection of protocols, problems with the various implementations of ASN.1/PER encoding/decoding, and the fact that these protocols—alone and in concert—have not endured the same level of scrutiny that other more common protocols have been subjected to. For example, we have unpublished data that shows that flooding a gateway or media server with GRQ request packets (RAS registration request packets) results in a DoS against certain vendor gateway implementations—basically the phones deregister.

SIP-Specific Attacks

Multiple vendors have confirmed vulnerabilities in their respective SIP (Session Initiation Protocol) implementations. The vulnerabilities have been identified in the INVITE message used by two SIP endpoints during the initial call setup. The impact of successful exploitation of the vulnerabilities has not been disclosed but potentially could result in a compromise of a vulnerable device. (CERT: CA-2003-06.) In addition, many recent examples of SIP Denial of Service attacks have been reported.

Recent issues that affect Cisco SIP Proxy Server (SPS) [Bug ID CSCec31901] demonstrate the problems SIP implementers may experience due to the highly modular architecture or this protocol. The SSL implementation in SPS (used to secure SIP sessions) is vulnerable to an ASN.1 BER decoding error similar to the one described for H.323 and other protocols. This example illustrates a general concern with SIP: As the SIP protocol links existing protocols and services together, all the classic vulnerabilities in services such as SSL, HTTP, and SMTP may resurface in the VoIP environment.

Summary

DoS attacks, whether they are intentional or unintended, are the most difficult VoIP-related threat to defend against. The packet switching nature of data networks allows multiple connections to share the same transport medium. Therefore, unlike telephones in circuit-switched networks, an IP terminal endpoint can receive and potentially participate in multiple calls at once. Thus, an endpoint can be used to amplify attacks. On VoIP networks, resources such as bandwidth must be allocated efficiently and fairly to accommodate the maximum number of callers. This property can be violated by attackers who aggressively and abusively obtain an unnecessarily large amount of resources. Alternatively, the attacker simply can flood the network with large number of packets so that resources are unavailable to all other callers.

In addition, viruses and worms create DoS conditions due to the network traffic generated by these agents as they replicate and seek out other hosts to infect. These agents are proven to wreak havoc with even relatively well-secured data networks. VoIP networks, by their nature, are exquisitely sensitive to these types of attacks. Remedies for DoS include logical network partitioning at layers 2 and 3, stateful firewalls with application inspection capabilities, policy enforcement to limit flooded packets, and out-of-band management. Out-of-band management is required so that in the event of a DoS event, system administrators are still able to monitor the network and respond to additional events.

Theft of services and information is also problematic on VoIP networks. These threats are almost always due to active attack. Many of these attacks can be thwarted by implementing additional security controls at layer 2. This includes layer 2 security features such as DHCP Snooping, Dynamic ARP Inspection, IP Source Guard, Port Security, and VLAN ACLs. The fundamental basis for this class of attacks is that the identity of one or more of the devices that participate is not legitimate.

Endpoints must be authenticated, and end users must be validated in order to ensure legitimacy. Hijacking and call interception revolves around the concept of fooling and manipulating weak or nonexistent authentication measures. We are all familiar with different forms of authentication, from the password used to login to your computer to the key that unlocks the front door. The conceptual framework for authentication is made up of three factors: "something you have" (a key or token), "something you know" (a password or secret handshake), or "something you are" (fingerprint or iris pattern). Authentication mechanisms validate users by one or a combination of these. Any type of unauthenticated access, particularly to key infrastructure components such as the IP PBX or DNS server, for example, can result in disagreeable consequences for both users and administrators.

VoIP relies upon a number of ancillary services as part of the configuration process, as a means to locate users, manage servers and phones, and to ensure favorable transport, among others. DNS, DHCP, HTTP, HTTPS, SNMP, SSH, RSVP, and TFTP services all have been the subject of successful exploitation by attackers. Potential VoIP users may defer transitioning to IP Telephony if they believe it will reduce overall network security by creating new vulnerabilities that could be used to compromise non-VoIP systems and services within the same network. Effective mitigation of these threats to common data networks and services could be considered a security baseline upon which a successful VoIP deployment depends. Firewalls, network and system intrusion detection, authentication systems, anti-virus scanners, and other security controls, which should already be in place, are required to counter attacks that might debilitate any or all IP-based services (including VoIP services).

H.323 and SIP suffer security vulnerabilities based simply upon their encoding schemes, albeit for different reasons. Because SIP is an unstructured text-based protocol, it is impossibly to test all permutations of SIP messages during development for security vulnerabilities. It's fairly straightforward to construct a malformed SIP message or message sequence that results in a DoS for a particular SIP device. This may not be significant for a single UA endpoint, but if this "packet of death" can render all the carrier-class media gateway controllers in a network useless, then this becomes a significant problem. H.323 on the other hand is encoded according to ASN.1 PER encoding rules. The implementation of H.323 message parsers, rather than the encoding rules themselves, results in security vulnerabilities in the H.323 suite.