On the Feasibility of Launching the Man-In-The-Middle Attacks on VoIP from Remote Attackers

Ruishan Zhang[†], Xinyuan Wang[†], Ryan Farley[†], Xiaohui Yang[†], Xuxian Jiang[‡]

[†]Department of Computer Science
George Mason University
Fairfax, VA 22030, USA
{rzhang3, xwangc, rfarley3, xyang3}@gmu.edu

[‡]Department of Computer Science
N.C. State University
Raleigh, NC 27606, USA
jiang@cs.ncsu.edu

ABSTRACT

The man-in-the-middle (MITM) attack has been shown to be one of the most serious threats to the security and trust of existing VoIP protocols and systems. For example, the MITM who is in the VoIP signaling and/or media path can easily wiretap, divert and even hijack selected VoIP calls by tempering with the VoIP signaling and/or media traffic. Since all previously identified MITM attacks on VoIP require the adversary initially in the VoIP signaling and/or media path, there is a common belief that it is infeasible for a remote attacker, who is not initially in the VoIP path, to launch any MITM attack on VoIP. This makes people think that securing all the nodes along the normal path of VoIP traffic is sufficient to prevent MITM attacks on VoIP.

In this paper, we demonstrate that a remote attacker who is not initially in the path of VoIP traffic can indeed launch all kinds of MITM attacks on VoIP by exploiting DNS and VoIP implementation vulnerabilities. Our case study of Vonage VoIP, the No.1 residential VoIP service in the U.S. market, shows that a remote attacker from anywhere on the Internet can stealthily become a remote MITM through DNS spoofing attack on a Vonage phone, as long as the remote attacker knows the phone number and the IP address of the Vonage phone. We further show that the remote attacker can effectively wiretap and hijack targeted Vonage VoIP calls after becoming the remote MITM. Our results demonstrate that (1) the MITM attack on VoIP is much more realistic than previously thought; (2) securing all nodes along the path of VoIP traffic is not adequate to prevent MITM attack on VoIP; (3) vulnerabilities of non-VoIP-specific protocols (e.g., DNS) can indeed lead to compromise of VoIP.

Categories and Subject Descriptors

C.2.0 [**Computer-Communication Networks**]: General—*Security and protection (e.g., firewalls)*; C.2.3 [**Computer-Communication Networks**]: Network Operations—*Network monitoring*

ASIACCS'09, March 10–12, 2009, Sydney, NSW, Australia.
Copyright 2009 ACM 978-1-60558-394-5/09/03 ...$5.00.

General Terms

Security, Reliability

Keywords

VoIP Security, SIP, MITM Attacks, DNS Spoofing

1. INTRODUCTION

VoIP has experienced explosive growth in the past few years, and it is becoming an indispensable part of more and more people's daily life. An IDC report [4] predicted that the number of U.S. residential VoIP subscribers will reach 44 million by 2010. In addition, VoIP has been widely used for carrying mission critical 911 calls. The Federal Communications Commission (FCC) estimated [2] that there were about 3.5 million residential VoIP 911 calls in 2006. Therefore, failures in providing reliable and trustworthy VoIP services not only disrupt the the normal operation of our society but also may cost people's lives under certain circumstances.

VoIP is built upon the interaction of a number of application protocols on the Internet. The open architecture of the Internet, however, makes VoIP protocols subject to more attacks than what is possible in PSTN (public switched telephone network). Signaling protocol and media transport protocol are two integral components of any VoIP system. Currently, the Session Initiation Protocol (SIP) [20] and the Real Time Transport Protocol (RTP) [22] are the dominant VoIP signaling protocol and media transport protocol respectively. In fact, most deployed VoIP services (e.g., Vonage, AT&T, Gizmo and Wengophone) use SIP and RTP. In addition, all existing VoIP systems depend on DNS to function normally. Therefore, any vulnerabilities in SIP, RTP or DNS could lead to the compromise of VoIP security and trustworthiness.

Previous research [20, 12, 1, 26, 24, 15] has shown that a man-in-the-middle (MITM), who is in the path of VoIP traffic, is able to wiretap, divert and even hijack selected VoIP calls by tempering with the VoIP signaling and/or media traffic. Such MITM attacks on VoIP could cause serious consequences to the targeted VoIP users. For example, VoIP wiretapping enables attackers to collect sensitive information (e.g., credit card number, bank account number, PIN) of the victim VoIP users. Unauthorized VoIP call diversion and voice pharming [24] could trick even the most meticulous VoIP callers into talking with bogus bank teller or interacting with bogus interactive voice response (IVR) systems. All these MITM attacks on VoIP could cause identity theft and financial loss to the victim VoIP users.

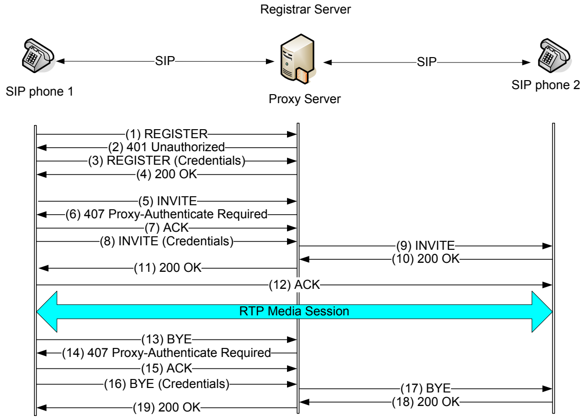


Figure 1: An Example of Message Flow of SIP Authentication

Since all previously identified MITM attacks on VoIP require the adversary initially in the VoIP signaling and/or media path, there is a common belief that it is infeasible for a remote attacker, who is not initially in the VoIP path, to launch any MITM attack on VoIP. As a result, many people do not believe the MITM attack is a realistic threat to current VoIP protocols and systems and they think that securing all the nodes along the normal path of VoIP traffic is sufficient to prevent MITM attacks on VoIP.

In this paper, we investigate the feasibility for a remote attacker, who is not initially in the path of VoIP traffic, to become the MITM. Our case study of Vonage VoIP service, which is the No. 1 residential VoIP service in the U.S. [9], shows that a remote attacker from anywhere on the Internet can, by exploiting the vulnerabilities of DNS and SIP message handling in the Vonage phone, stealthily become the remote MITM and launch all kinds of MITM attacks on target VoIP phones. Specifically, we find that

- the remote attacker can crash and reboot the targeted Vonage SIP phone by sending it crafted, malformed SIP `INVITE` messages. This will cause the rebooted Vonage SIP phone to send out DNS query about the location of the SIP server to contact.
- the remote attacker can trick the Vonage SIP phone into taking any IP address as that of the Vonage SIP server via spoofed DNS responses.
- the remote attacker can cause all the calls to or from the targeted Vonage phone to pass it. This makes the remote attacker a MITM and enables him to wiretap and hijack any calls to or from the targeted Vonage phone.

Note, the identified remote MIMT attack on VoIP only requires the knowledge of the phone number and the IP address of the targeted Vonage phone, and it works even if the targeted Vonage phone is behind NAT.

Our results demonstrate that (1) the MITM attack on VoIP is much more realistic than previously thought; (2) securing all nodes along the path of VoIP traffic is not adequate to prevent MITM attack on VoIP; (3) vulnerabilities of non-VoIP-specific protocols (e.g., DNS) can indeed lead to compromise of VoIP.

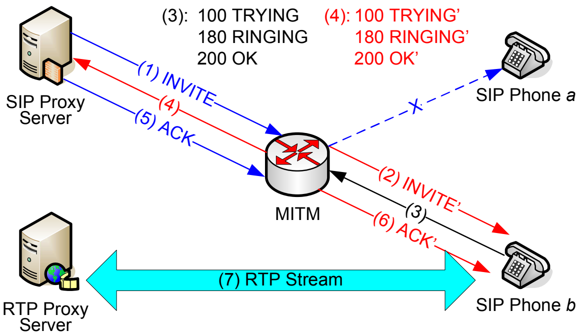


Figure 2: Unauthorized Call Redirection via MITM

The rest of this paper is organized as follows. Section 2 gives a brief overview of SIP and the MITM attack. Section 3 describes our investigation approach. Section 4 presents our case study and demonstrates the DNS spoofing, wiretapping and call hijacking attacks on a Vonage SIP phone. Section 5 discusses potential mitigation strategies. Section 6 reviews related work. Finally, section 7 concludes the paper.

2. OVERVIEW OF SIP AND THE MIMT ATTACK

SIP is a HTTP-like, application layer signaling protocol used to create, modify, and terminate multimedia sessions (e.g., VoIP calls) among Internet endpoints. The SIP specification defines the following different components: user agents (UA), proxy servers, redirect servers, registrar servers, location servers. An UA represents an endpoint of the communication (i.e., a SIP phone). The proxy server is the intermediate server that forward the SIP messages from UAs to its destination. Various SIP servers described above are logical functions. In most deployed systems, generic SIP servers perform the functionalities of both registrar servers and proxy servers.

The SIP specification [20] recommends using TLS or IPSec to protect SIP signaling messages, and using S/MIME to protect the integrity and confidentiality of SIP message bodies. However, most deployed SIP VoIP systems (e.g., Vonage, AT&T CallVantage) only use SIP authentication to protect their signaling messages.

SIP authentication is similar to digest based HTTP authentication. Figure 1 depicts the typical SIP authentication of call registration, call setup and call termination. When a SIP server (e.g., proxy, registrar) receives a SIP request (e.g., `REGISTER`, `INVITE`, `BYE`) from a SIP phone, the SIP server challenges the SIP phone with either a `401 unauthorized` or a `407 proxy-authentication required` message. Upon receiving the `401` or `407` message, the SIP phone calculates a hash value by applying a specific digest algorithm (e.g., MD5) to SIP message fields *request-URI*, *username*, *shared password between the phone and the SIP server*, *realm*, and *nonce*. Then the SIP phone sends the hash value along with the original SIP request as the authentication credential.

However, existing SIP authentication only covers selected fields of a few SIP messages from a SIP phone to a SIP server. This leaves other SIP messages and fields unprotected. By exploiting the vulnerabilities of SIP and RTP, a MITM who is in the path of VoIP traffic can

- detour any chosen call via anywhere on the Internet [24]. This would allow the attacker to wiretap selected VoIP calls and collect sensitive information (e.g., account number, PIN) from the victim.

- redirect any selected VoIP call to any third party and manipulate and set the call forwarding setting of any selected Gizmo VoIP subscriber without authorization [24]. This would allow the attacker to hijack VoIP calls to financial institutions and pretend bank representative.

- launch billing attacks [26] on selected VoIP users such that the victim VoIP users will either be overcharged for their VoIP calls or charged for calls not made by them.

- disrupt any chosen VoIP call by sending a `BUSY` or `BYE` message.

Figure 2 illustrates the message flow of the unauthorized call redirection attack by the MITM. All existing MITM attacks require the attacker initially in the VoIP signaling and/or media path, this somewhat limiting requirement makes many people believe that the MITM attack on VoIP is not realistic. In the following sections, we investigate how a remote attacker, who is not initially in the VoIP path, can become the remote MITM and launch all kinds of MITM attacks on targeted VoIP users.

3. INVESTIGATION APPROACH

To investigate the feasibility for the remote attacker to become the MITM of VoIP traffic, we assume the role of the active adversary who seeks to trick the targeted VoIP phone to pass all its VoIP traffic through him by exploiting the vulnerabilities of the SIP phone and all protocols it uses. We choose to experiment with Vonage VoIP, which is the most popular residential VoIP service in the U.S. market.

Our investigation is divided into two steps. First, we passively observe the network traffic between our Vonage SIP phone and its VoIP servers to spot potential weaknesses. Second, we use fuzz testing to confirm the weaknesses found by passive observation or identify new possible flaws. Note that we treat the VoIP phone as a whole, and look for all the vulnerabilities from the embedded operating system and the upper-layer applications. When observing the network traffic, we use Wireshark [11] to view the parsed protocols .

By observing the network traffic, we found a weakness of the Vonage phone in handling DNS. A Vonage SIP phone obtains SIP servers' IP addresses via DNS query [18]. Given that DNS runs over connectionless UDP, the remote attacker can forge and inject DNS response packets to the SIP phone. Whether the victim accepts the forged DNS response depends on whether the following conditions are satisfied:

- The destination IP address and the destination port number of the forged DNS response packet are the source IP address and the source port number of the DNS query packet.

- The source IP address and the source port number of the forged DNS response packet are the destination IP address and the destination port number of the DNS query packet.

- The ID field of the forged DNS response packet matches that of the DNS query packet.

- The question section of the forged DNS response packet matches the question section of one of the DNS query packets sent.

Since both the ID and the port number are 16 bits, the whole brute-force search space for a matching DNS response should be 2^{32} in theory. However in practice, if a DNS query uses predictable IDs and/or a limited port range, the brute-force search space could be greatly reduced. One key finding of our research is that the Vonage SIP phone uses a static ID and a small range of port number 45000-46100, which reduces the brute-force search space to merely 1100.

In order to trick the targeted SIP phone to accept the spoofed DNS response, the remote attacker needs to trigger a DNS query from the targeted SIP phone. We have observed that the SIP phone sends a DNS query each time it restarts. Therefore, if the remote attacker can somehow cause the target SIP phone to reboot, he can reach this goal. After a lot of fuzz testing, we have identified a program flaw in handling a malformed `INVITE` message, which allows the remote attacker to remotely crash and reboot the Vonage SIP phone, thus triggering a DNS query.

Utilizing the above vulnerabilities and techniques, a remote attacker is able to inject fake DNS responses to the Vonage SIP phone and trick it into thinking that the remote attacker is the Vonage SIP server. By replacing the the source IP address of the `REGISTER` message from the Vonage SIP phone with its own IP address, the remote attack can make the Vonage server into thinking it is the Vonage SIP phone. As a result, the remote attacker becomes a MITM on the path between the SIP phone and its SIP servers.

Our implementation of the remote attacks consist of approximately 6000 lines of C code. Logically, it consists of three parts: (1) the remote MITM module which let any remote attacker become the remote MITM by crashing the targeted SIP phone and injecting the spoofed DNS responses; (2) the remote wiretapping module that allows the remote MITM to wiretap selected VoIP calls; (3) the remote call hijacking module that allows the remote MITM to hijack selected VoIP calls.

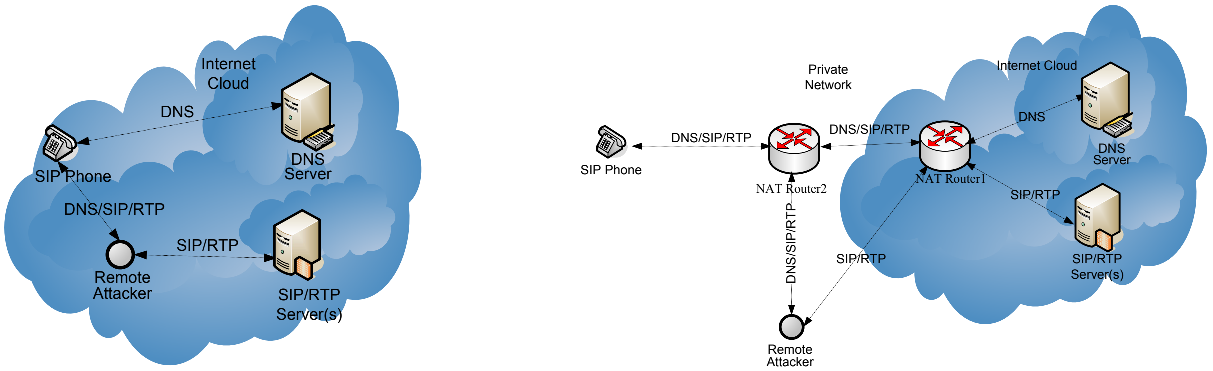
4. CASE STUDY

In this section, we describe our case study of Vonage VoIP service, which is the No.1 U.S. residential VoIP service with more than 2.5 million subscribers. Note all our exploiting experiments have been against our own phones and account.

We demonstrate how a remote attacker becomes a MITM by launching DNS spoofing attack on a Vonage SIP phone. First we describe our testbed setup and message flow of the normal startup or reboot of the Vonage SIP phone. Then we present the identified DNS implementation weaknesses of the Vonage phone and its vulnerability in handling the malformed `INVITE` message. Next we illustrate the message flow of the DNS spoofing attack and describe our experimental results. Finally after achieving a MITM, we present the remote wiretapping and remote call hijacking attacks on VoIP.

4.1 Network Setup

Figure 3 illustrates the network setup of our testbed. The remote attacker runs Red Hat Linux on a Dell D610 laptop



(a) SIP phone directly connected to the Internet

(b) SIP phone behind NATs

Figure 3: Testbed Setup

computer. NAT router 1 is a FreeBSD machine running on a virtual machine and NAT router 2 is a Linksys router.

Figure 3(a) illustrates the network setup where the SIP phone is directly connected to the Internet. We use SIP/RTP server(s) to denote the SIP server and the RTP server which handle the signaling messages and the RTP stream respectively. The remote attacker could be anywhere on the Internet. In our experiment, we use a wiretap device to capture live network traffic transited from/to the SIP phone. The wiretap device and the SIP phone connect to a four port 10BASE-T Ethernet hub.

Figure 3(b) illustrates the network setup where the SIP phone is behind NATs. Note this setup is different from the most popular settings where the SIP phone is behind only one NAT router. We notice that the SIP phone will send some destination unreachable ICMP packets to the Vonage DNS server when receiving spoofed DNS responses with unmatched port numbers. We use the NAT router2 to block these unwanted traffic from reaching the Vonage DNS server.

As a result, the SIP phone is behind 2 NAT routers. For convenience, we placed the remote attacker outside NAT router2 but inside the private network of NAT router1. From the remote attacker's perspective, the targeted SIP phone is behind one NAT router, which is the most likely configuration for residential VoIP phones. In this configuration, the wiretap device and NAT Router2 connect to a four port 10BASE-T Ethernet hub. We notice that none of the NAT router will change the source port number of the passing packet, this enables the remote attacker to become the remote MITM via the identified exploit even if the targeted Vonage phone is behind 2 levels of NAT routers.

4.2 Message Flow of Normal Startup or Reboot

Figure 4 depicts the message flow of normal startup or reboot of a Vonage phone. At the beginning, the SIP phone sends a DNS query to the Vonage DNS server to ask for SIP servers's IP addresses in step (1). All DNS queries from the Vonage SIP phone go to the Vonage DNS server at IP address 216.115.31.140. Then in step (2), the Vonage DNS server replies with a DNS response packet containing four IP addresses of Vonage SIP servers: 69.59.252.35, 69.59.232.42, 69.59.242.84 and 69.59.227.87. At step (3), the Vonage phone sends to one of four SIP servers a SIP `REGISTER` message. Then in step (4), the SIP server challenges the SIP phone with a `401 Unauthorized` message. After receiving the `401 response`, the SIP phone sends the SIP server a new SIP `REGISTER` message containing credentials. Note the ”expires” field in the SIP `REGISTER` message specifies the duration for which this registration will be valid. So the SIP phone needs to refresh its registration from time to time.

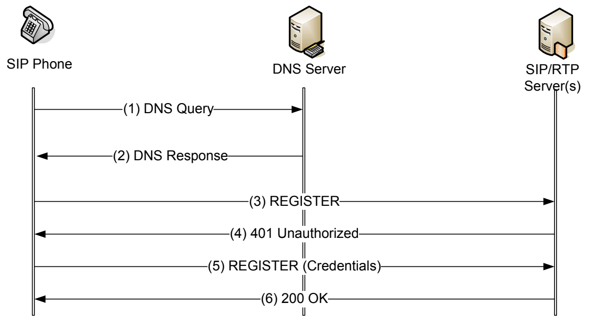


Figure 4: Message Flow of Normal Startup or Reboot

4.3 Exploitable Vulnerabilities of Vonage SIP Phone

4.3.1 Weaknesses in the Implementation of DNS Query and Response

The implementation of DNS query/response in the Vonage phone has several weaknesses.

- The SIP phone always uses a static ID value, 0x0001, in all DNS queries.
- The source port number range of DNS queries is limited to 45000-46100.
- The question sections of all DNS queries are identical, and contain 11 bytes of string `d.voncp.com`.
- The SIP phone does not check the source IP address of a DNS response. Even if the source IP address is not that of the Vonage DNS server, the Vonage phone still accepts a spoofed DNS response.

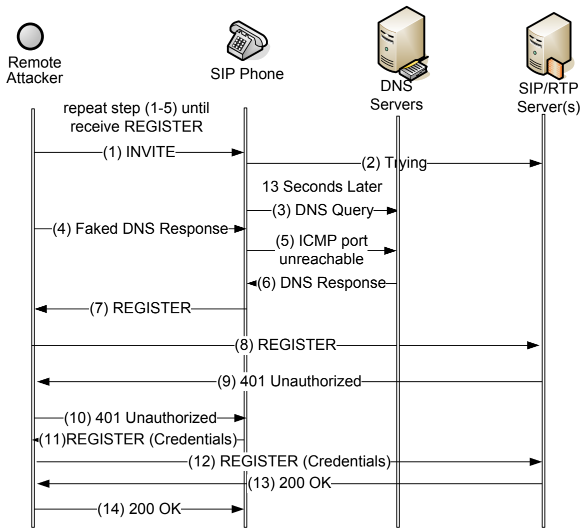


Figure 5: Message Flow of DNS Spoofing Attack

Due to these vulnerabilities, the brute-force search space for forging a matching DNS response is no more than 1100.

4.3.2 *Vulnerability in Handling Malformed INVITE Messages*

We have found that our Vonage SIP phone fails to handle a malformed `INVITE` message correctly and it will reboot when receives a malformed `INVITE` message with a over length phone number in the `From` field. This allows the remote attacker to crash and reboot the targeted Vonage phone by sending it one malformed `INVITE` message. To launch such an attack, the remote attacker needs to spoof the source IP address as that of one of Vonage SIP servers. Otherwise, the Vonage phone will discard the `INVITE` message. Our experiments have shown that the Vonage phone does not ring but replies with a `Trying` message after receiving the malformed `INVITE` messages. Then the phone crashes and reboots almost immediately. After a few seconds (e.g., 13 seconds), the Vonage phone sends a DNS query to the Vonage DNS sever. Note the SIP phone crash attack is stealthy in that the SIP phone does not ring at all when receives the malformed `INVITE` message.

4.4 DNS Spoofing Attack

4.4.1 *Message Flow*

Figure 5 shows the SIP message flow of the DNS spoofing attack on the Vonage SIP phone. At the beginning, the remote attacker sends a malformed `INVITE` message to the SIP phone with a spoofed source IP in step (1). In response, the SIP phone sends a `Trying` message to the real SIP server in step (2). Then the SIP phone crashes and reboots. Several seconds later, the SIP phone sends a DNS query to the Vonage DNS server asking for the SIP servers' IP addresses in step (3). Within several milliseconds, the legitimate DNS response from the Vonage DNS server reaches the SIP phone in step (6).

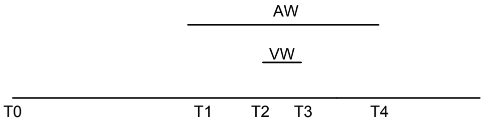


Figure 6: Timeline of a Round of Attack

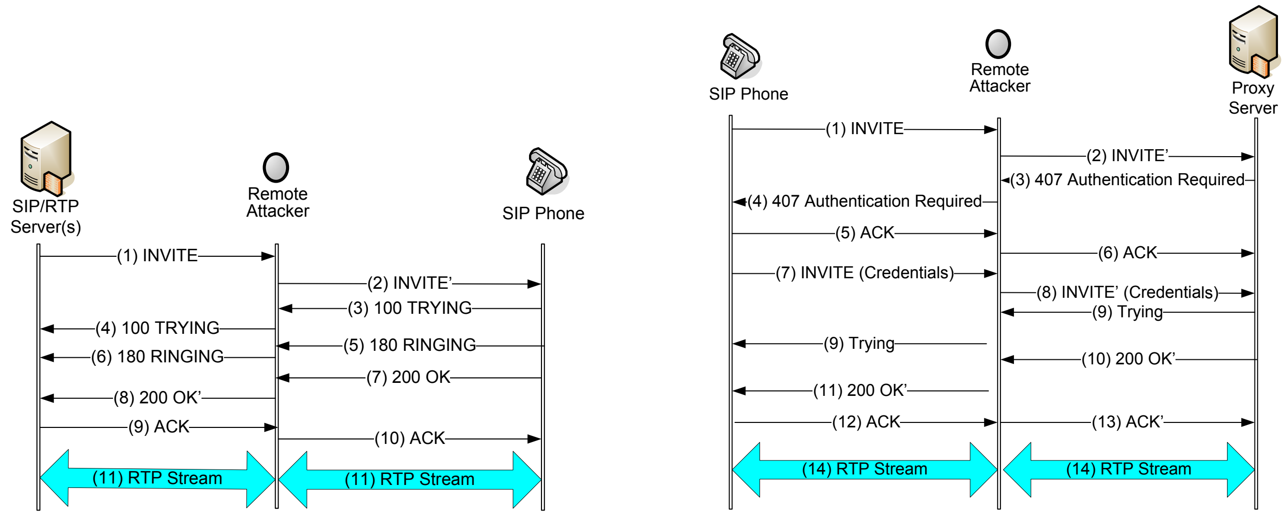
If the remote attacker sends the spoofed DNS response packets to the Vonage phone within the time window from step (3) to (6), the Vonage phone will receive the spoofed DNS response before the legitimate DNS response arrives. This process is represented at step (4). Since the remote attacker does not have access to the original DNS query from the Vonage phone, he has to try each of the 1100 possible port numbers in the spoofed DNS response packets. If the spoofed DNS response packet contains the wrong port number, the Vonage phone sends a port unreachable ICMP packet to the DNS server at step (5). If the spoofed DNS response packet contains the matching port number, the Vonage phone accepts the spoofed DNS response packet and sends out `REGISTER` message to the remote attacker at step (7) as it now thinks the remote attacker is the Vonage SIP server. Therefore, the remote attacker can determine the success of the DNS spoofing by checking if he receives the expected `REGISTER` from the targeted Vonage phone within a predefined period of time.

If the remote attacker does not receive the expected `REGISTER` from the targeted Vonage phone within predefined period of time, he knows that the Vonage phone has accepted the authentic DNS response from the Vonage DNS server. The remote attacker needs to start a new round of attack by repeating steps (1-6) until he receives a `REGISTER` message from the SIP phone in step (7). We define steps from (1) to (6) as a round of the attack. Normally it will take several rounds before the SIP phone finally sends the `REGISTER` message to the remote attacker.

After receiving the `REGISTER` message at step (7) or (11), the remote attacker forwards them to the real SIP server in step (8) or (12). Meanwhile the remote attacker forwards the `401 Unauthorized` message at step (9) and the `200 OK` message at step (13) from the SIP server to the SIP phone in step (10) and (14). Now the remote attacker becomes the MITM in that 1) the SIP phone thinks the remote attacker is the SIP server; and 2) the SIP server thinks the remote attacker is the SIP phone.

To launch the DNS spoofing attack, the remote attacker only need to construct 1000 fake DNS response packets with 1000 different destination port numbers. Specifically, the remote attacker just need to

- Fill 0x0001 into the ID field of all spoofed DNS responses.
- Fill `d.voncp.com` into the question section of all DNS responses.
- Fill the IP address of the remote attacker into the answer section of all spoofed DNS responses.
- Set the destination port number of 1st, 2nd,..., 1000th packet as 45000,45001,...,45999.
- The SIP phone does not check source IP address. So we set it to the IP address of the remote attacker when the victim phone is on the Internet. When the phone is behind NATs, the source IP address of spoofed DNS packets is set to that of Vonage SIP server to pass through NAT Router2.



(a) PSTN Phone Calls SIP Phone

(b) SIP Phone Calls PSTN Phone

Figure 7: Message Flow of Wiretapping Calls Between a SIP Phone and a PSTN Phone by the Remote Attacker

Table 1: Measured Time Interval from `INVITE` to DNS Query without Spoofed DNS

10times	1	2	3	4	5	6	7	8	9	10	range	average
seconds	14.9	13.8	13.0	18.8	14.6	12.9	15.5	12.8	15.5	14.1	12.9-15.5	14.9

Figure 6 illustrates the timeline of a round of the attack. T0 is the time when the remote attacker sends a malformed `INVITE`. T2 and T3 are the times when the SIP phone sends a DNS query and receives the legitimate response from the DNS server respectively. We refer to the time interval from T2 to T3 as the Vulnerable Window (VM). T1 and T4 denote the start time and end time respectively of sending spoofed DNS response packets. We refer to the time interval from T1 to T4 as an Attack Window (AW). Apparently, the larger the attack window is, the fewer rounds the remote attacker needs in order to succeed.

Our experiments show that the Vonage phone actually accepts spoofed DNS response before it sends out the DNS query. In addition, if the remote attacker keeps sending many spoofed DNS response packets with very shot inter-packet arrival time, it will have a good chance to block the targeted SIP phone from receiving the authentic DNS response. Therefore, the attack window could start earlier and end later than the vulnerable window.

4.4.2 Experimental Results and Analysis

Ideally we want T1 to be earlier but not too much earlier than T2. We have measured the time interval from the moment the remote attacker sends the malfored `INVITE` message to the moment the crashed and reboot SIP phone sends the first DNS query. Table 1 shows the measured the time intervals for 10 runs of crashing the SIP phones. It shows that it takes 12.9 $\sim$ 15.5 seconds for the SIP phone to send the first DNS query after receiving the malformed `INVITE` packet. Therefore, we set T1 at 12 seconds after T0. We have set transmission rate of the spoofed DNS response packets at 1000 pkt/s. To maximize the chance of hitting the correct port number while keeping the the duration of each round short, we set the duration of attack window to be 8 seconds. Therefore, T4 is 20 seconds after T0. At each round, the remote attacker sends the 1000 spoofed DNS response packets for maximum 8 times, and the duration of one round of attack is 20 seconds. As shown in Table 2, the average number of rounds and the required time of 10 instances of DNS spoofing attack against the SIP phone on the Internet is 39.8 and 789 seconds (about 13 minutes).

When the SIP phone is behind NATs, the attack is similar except that the IP address of fake DNS responses should be spoofed as that of the Vonage DNS server to pass through NAT Router2. The result of one test showed that the number of rounds is 8, and the required time is 169 seconds.

Our preliminary investigation shows that port numbers of DNS queries are all in the range 45000-45999, so that the range 45000-45999 is applied.

The packet size of a spoofed DNS response is 87 bytes, including 14 bytes of Ethernet header, 20 bytes of IP header, 8 bytes of UDP header and 45 bytes of UDP payload. Given that the DNS spoofed packets are transmitted at 1000 pkt/s, the transmission rate is about 700 kbps. Since most household broadband Internet access has at least than 2 Mbps downstream rate, our DNS spoofing is practically applicable to household broadband VoIP.

4.5 Wiretapping and Call Hijacking

After becoming a MITM, the remote attacker is able, at least in theory, to launch all kinds MITM attacks. In this subsection, we demonstrate two representative MITM attacks from the remote attacker: call wiretapping and call hijacking.

4.5.1 Wiretapping Incoming Call Remotely

Table 2: Number of Rounds and Time Needed to Become the Remote MITM

10 instances	1	2	3	4	5	6	7	8	9	10	Sum	Average
#round	7	9	11	15	22	28	41	54	105	106	398	39.8
Time (sec)	135	175	213	296	437	556	800	1080	2081	2117	7890	789

Figure 7(a) shows the message flow of wiretapping the incoming calls to the Vonage phone by the remote attacker.

At the beginning, the SIP server sends an `INVITE` message to the remote attacker at step (1). The remote attacker modifies the IP address and port number for the upcoming RTP stream in the `INVITE` message so that upcoming RTP stream from the SIP phone will go to the remote attacker's IP address and port number 12345. Then the remote attacker sends the modified `INVITE` message to the SIP phone at step (2). At step (3-6), the remote attacker forwards `Trying` and `Ringing` messages from the SIP phone to the SIP server. After the receiver picks up the phone, the SIP phone sends a `200 OK` message at step (7) to the remote attacker. Similar to step (2), the remote attacker sets its own IP address and port number (e.g., 12345) as the RTP stream termination point, and then sends the modified `200 OK` to the SIP server at step (8). At step (9-10), the remote attacker forwards the `ACK` message from the SIP server to the SIP phone. At this point, the three way handshake for the VoIP call setup is completed. Then at step (11), the remote attacker wiretaps the RTP streams between the SIP phone and the RTP server as the remote MITM.

4.5.2 *Wiretapping Outgoing Call Remotely*

Figure 7(b) illustrates the message flow of wiretapping the outgoing calls from the Vonage phone by the remote attacker.

At the beginning, the SIP phone sends an `INVITE` message to the remote attacker at step (1). Then the remote attacker modifies the IP address and port number for the upcoming RTP stream and sends the modified `INVITE` message to the SIP server at step (2). At step (3-4), the remote attacker forwards the `407 proxy-authentication Required` message to the SIP phone. At step (5-6), the remote attacker forwards the `ACK` message for `407 proxy-authentication Required` to the SIP server. At step (7), the SIP phone sends a new `INVITE` message with the required credential to the remote attacker. Simila to step (2), the remote attacker modifies the IP address and port number for the upcoming RTP and sends the modified `INVITE` message to the SIP server at step (8). At step (9-10), the remote attacker forwards the `Trying` message to the SIP phone. At step (10), the SIP server sends a `200 OK` message to the remote attacker. Similar to step (8) in Figure 7(a), the remote attacker modifies RTP termination information and sends the `modified 200 OK` message to the SIP phone. At step (12), the SIP phone sends an `ACK` message to the remote attacker, who modifies the RTP termination information and forward the modified `ACK` message to the SIP server at step (13). At step (14), the remote attacker wiretaps RTP the streams between the SIP phone and the RTP server as the remote MITM.

4.5.3 *Call Hijacking Attack*

Figure 8 illustrates the message flow of call hijacking by the remote attacker. When a VoIP user dials a PSTN phone number from the SIP phone, the SIP phone sends an `INVITE` message to the remote attacker at step (1). The remote attacker responds with a spoofed `200 OK` message at step (2). Then the SIP phone accepts the spoofed `200 OK` message, and responds with `ACK` message to the remote attacker at step (3) to finish the three way handshake. At step (4) the caller talks to the remote attacker while thinking he is talking to the intended callee.

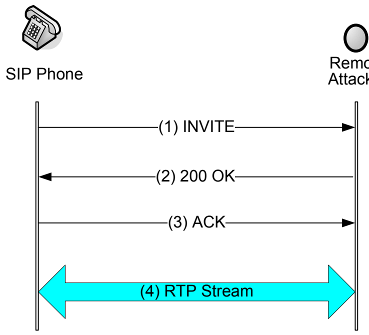


Figure 8: Message Flow of Call Hijacking

5. DISCUSSIONS

Using the techniques of passive observation and active fuzz testing, we have demonstrated how a remote attacker can become a MITM by exploiting the design and implementation flaws in VoIP phones. While our spoofing attack exploits specific weaknesses in a specific VoIP system, the investigation approach could be applied to any VoIP systems. In fact, we have applied our fuzz testing on an AT&T SIP phone, and we have found that a remote attacker can crash the AT&T SIP phone by sending it a malformed SIP message. Our experimental results have further shown that other VoIP phones (e.g., Wengophone) also have exploitable implementation bugs. Therefore, many existing deployed VoIP phones could be vulnerable to the newly identified remote MITM attack.

To fix the VoIP phone weaknesses identified in this paper, first the SIP phone should correctly check the validity of the phone number in the `INVITE` message, which would prevent the remote attacker from crashing and rebooting the SIP phone using the identified malformed `INVITE` message. Second, the SIP phone should should use randomly selected 16 bit ID and and 16 bit source port number. This would increase the brute force search space for a matching DNS response to 2^{32} and make it infeasible for a remote attacker to spoof the DNS response. In addition, the SIP phone should always check if the source IP address of a DNS response is that of the known DNS server.

While it is easy to fix the specific flaws identified in this paper, it is almost impossible to make VoIP phones and the implementations of VoIP protocols bug-free. To prevent remote attackers from exploiting other potential weaknesses of VoIP phone and protocols, we suggest the following strategies

- Use SSL/TLS and SRTP to protect SIP messages and voice RTP streams whenever possible. Note the implementation of SSL/TLS or SRTP might introduce new exploitable vulnerabilities. For example, the implementation of Openssl versions 0.98 is vulnerable to a remote heap overflow exploit, which could cause arbitrary code executed [5]. Our experiments show that none of the major commercial VoIP services (e.g., Vonage, AT&T, broad voice) uses SSL or SRTP for the VoIP traffic between the VoIP phone and the VoIP server. It might be worthwhile to investigate what has prevent the commercial service providers from deploying SSL/TLS and SRTP in their VoIP services.

- VoIP phones should have undergone extensive and indepth fuzz testing before being deployed. While this may not be able to discover all the exploitable vulnerabilities of the VoIP phone, it will at least raise the bar for the attacker to identify and exploit any vulnerability.

- It might be worthwhile to develop some light-weight VoIP intrusion detection system to be deployed at VoIP phone. For example, a VoIP traffic anomaly IDS should be able to detect our DNS spoofing attack when observing enormous DNS packets within a short period of time.

6. RELATED WORK

Most previous work investigated the threats and intrusion detection approaches for VoIP servers. Reynolds et al [19] proposed multi-protocol protection against flooding DoS attacks on VoIP servers. Wu et al [25] presented a cross protocol intrusion detection architecture to detect certain denial-of-service attacks on VoIP server. Sengar et al [23] proposed to utilize interactive protocol state machines to build intrusion detection systems. Dantu et al proposed a multi-stage spam filter based on trust and reputation [16]. Since all these methods are designed to protect VoIP servers side, they are unlikely effective against the attacks on end VoIP users.

Arkko et al [12] proposed a scheme to negotiate the security mechanism between two SIP entities. Baugher et al [13] proposed Secure Real-time Transport Protocol (SRTP) to protect the media traffic. However, currently SRTP is not widely being applied in deployed VoIP systems.

Salsano et al [21] evaluated the SIP processing overhead when SIP authentication and TLS are employed. Bellovin et al analyzed the challenges in applying the Communications Assistance to Law Enforcement Act (CALEA) to wiretap VoIP calls [14]. McGann and Sicker [17] analyzed detection capability of several VoIP security tools: SiVuS, PROTOS SIP Fuzzer [6], SIP Forum Test Framework and some commercial products. They showed that there exists large gap between known VoIP security vulnerabilities and the tool's detection capability. Zhang et al [26] empirically demonstrated that the Vonage and AT&T CallVantage were vulnerable to billing attacks. Wang et al [24] systematically studies the trust of current SIP-based VoIP and demonstrated a number of call diversion attacks on Vonage and AT&T VoIP users which can be used to launch voice pharming attacks on VoIP users.

Several previous work has explored the weaknesses in DNS [3] [7]. However, to the best of our knowledge, there is no published work on exploiting DNS weaknesses in deployed VoIP environments.

7. CONCLUSION

While the MITM attack on VoIP has been known for years, the feasibility of launching the MITM attack on deployed VoIP has been seriously underestimated since all previous MITM attacks require the adversary to be initially in the path of VoIP traffic.

The key contribution of this paper is that it demonstrates that the adversary does not have to be initially in the path of VoIP traffic to conduct the MITM attack. Our case study of Vonage VoIP service shows that a remote attacker can stealthily become a remote MITM and launch all kinds of MITM attacks (e.g., wiretap, call hijacking) as long as he knows the phone number and the IP address of the target VoIP phone. Our results demonstrate that (1) the MITM attack on VoIP is much more realistic than previously thought; (2) securing all nodes along the path of VoIP traffic is not adequate to prevent MITM attack on VoIP; (3) vulnerabilities of non-VoIP-specific protocols (e.g., DNS) can indeed lead to compromise of VoIP.

Acknowledgement

This work was partially supported by NSF Grants CNS-0524286 and CCF-0728771.

8. REFERENCES

[1] Black Hat USA 2007 Briefings. URL. http://www.blackhat.com/html/bh-usa-07/bh-usa-07-speakers.html

[2] First Report and Order and Notice of Proposed RuleMaking. URL. http://www.fcc.gov/cgb/voip911order.pdf.

[3] DNSSEC. URL. http://www.dnssec.net/.

[4] IDC Anticipates 34 Million More Residential VoIP Subscribers in 2010. URL. http://www.idc.com/getdoc.jsp?-containerId =prUS20211306.

[5] OpenSSL DTLS Implementation Remote Heap Overflow Vulnerability. URL. http://secwatch.org/advisories/1019254/

[6] PROTOS SIP Fuzzer. URL. http://www.ee.oulu.fi/research/ouspg/protos/testing /c07/sip/

[7] SANS Institute. DNS Spoofing by The Man In The Middle. http://www.sans.org/reading_room/whitepapers/dns /1567ṗhp

[8] Snort. URL. http://www.snort.org/

[9] US VoIP market shares. URL. http://blogs.zdnet.com/ITFacts/?p=11425.

[10] Vonage. URL. http://www.vonage.com/.

[11] Wireshark. URL. http://www.wireshark.org/

[12] J. Arkko, V. Torvinen, G. Camarillo, A. Niemi and T. Haukka. Security Mechanism Agreement for the Session Initiation Protocol (SIP). *RFC 3329, IETF*, January 2003.

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

[14] S. Bellovin, M. Blaze, E. Brickell, C. Brooks, V. Cerf, W. Diffie, S. Landau, J. Peterson and J. Treichler. Security Implications of Applying the Communications Assistance to Law Enforcement Act to Voice over IP. http://www.cs.columbia.edu/ smb/papers/ CALEAVOIPreport.pdf

[15] F. Cao and S. Malik. Vulnerability analysis and best practices for adopting IP telephony in critical infrastructure sectors. Communications Magazine, 44(4), Pages 138-145, April 2006.

[16] R. Dantu and P. Kolan. Detecting spam in VoIP networks. In *Proceedings of the Steps to Reducing Unwanted Traffic on the Internet Workshop (SRUTI 2005)*,Cambridge, MA, July 2005.

[17] S. McGann and D. C. Sicker. An analysis of Security Threats and Tools in SIP-Based VoIP Systems. *Second VoIP Security Workshop*, 2005.

[18] P. Mockapetris. Domain names - implementation and specification. *RFC 1035, IETF*, November 1987.

[19] B. Reynolds and D. Ghosal. Secure IP Telephony Using Multi-layered Protection In *Proceedgins of the 2003 Network and Distributed System Security Symposium (NDSS 2003)*, Feburary 2003.

[20] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M Handley and E. Schooler. SIP: Session Initiation Protocol. *RFC 3261, IETF*, June 2002.

[21] S. Salsano, L. Veltri, D. Papalilo. SIP Security Issues: the SIP Authentication Procedure and Its Processing Load. *IEEE Network*, 16(6), Pages 38–44, 2002.

[22] H. Schulzrinne, S. Casner, R. Frederick and V. Jacobson. RTP: A Transport Protocol for Real-Time Applications. *RFC 1889, IETF*, January 1996.

[23] H. Sengar, D. Wijesekera, H. Wang, and S. Jajodia. VoIP Intrusion Detection Through Interacting Protocol State Machines. In *Proceedgins of the 2006 International Conference on Dependable Systems and Networks (DSN 2006)*, June 2006.

[24] X. Wang, R. Zhang, X. Yang, X. Jiang, D. Wijesekera: Voice Pharming Attack and the Trust of VoIP. In *Proceedings of 4th International Conference on Security and Privacy in Communication Networks (SecureComm 2008)*, September 2008.

[25] Y. Wu, S. Bagchi, S. Garg, N. Singh. SCIDIVE: A Stateful and Cross Protocol Intrusion Detection Architecture for Voice-over-IP Environments In *Proceedgins of the 2004 International Conference on Dependable Systems and Networks (DSN 2004)*, Pages 433 – 442, July 2004.

[26] R. Zhang, X. Wang, X. Yang, X. Jiang. Billing Attacks on SIP-Based VoIP Systems. In *Proceedings of the First USENIX Workshop on Offensive Technologies (WOOT 2007)*, August 2007.