**NOTICE: This material is copyrighted, and I am required to inform you that its use is limited to permissions of the copyright holder of each particular manuscript. Please refer to the following list, sorted chronologically, for further information.**

- Ryan J. Farley and Errin W. Fulp. Effects of Processing Delay on Function-Parallel Firewalls. IASTED: PDCN February 2006.

  > The authors have transfered copyright in the Paper, transferring to IASTED and ACTA PRESS the exclusive right to publish, distribute, reproduce, or sell the Paper by printed, electronic or other means. the Paper cannot be re-published, distributed or sold ther than by IASTED and ACTA PRESS without the prior written permission of IASTED and ACTA PRESS. It is available for purchase at `http://www.actapress.com/Content_of_Proceeding.aspx?proceedingID=352`

- Errin W. Fulp and Ryan J. Farley. A Function-Parallel Architecture for High-Speed Firewalls. IEEE: ICC June 2006.

  > ©2006 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

- Ruishan Zhang, Xinyuan Wang, Ryan Farley, Xiaohui Yang, and Xuxian Jiang. On the Feasibility of Launching the Man-In-The-Middle Attacks on VoIP from Remote Attackers. ACM: ASIACCS March 2009.

  > Copyright ©2009 by the Association for Computing Machinery, Inc. (ACM). Permission to make digital or hard copies of portions of this work for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page in print or the first screen in digital media. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted.
  > To copy otherwise, to republish, to post on servers, or to redistribute to lists, requires prior specific permission and/or a fee. Send written requests for republication to ACM Publications, Copyright & Permissions at the address above or fax +1 (212) 869-0481 or email permissions@acm.org.
  > For other copying of articles that carry a code at the bottom of the first or last page, copying is permitted provided that the per-copy fee indicated in the code is paid through the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923.

- Ruishan Zhang, Xinyuan Wang, Xiaohui Yang, Ryan Farley and Xuxian Jiang. An Empirical Investigation into the Security of Phone Features in SIP-based VoIP Systems. ISPEC April 2009.

  > The copyright to the Contribution identied above is transferred to Springer-Verlag GmbH Berlin Heidelberg (hereinafter called Springer-Verlag). The copyright transfer covers the sole right to print, publish, distribute and sell throughout the world the said Contribution and parts thereof, including all revisions or versions and future editions thereof and in any medium, such as in its electronic form (ofine, online), as well as to translate, print, publish, distribute and sell the Contribution in any foreign languages and throughout the world (for U.S. government employees: to the extent transferable). The published article is available on Springers website `www.springerlink.com`.

- Ryan Farley and Xinyuan Wang. Roving Bugnet: Distributed Surveillance Threat and Mitigation. IFIP: SEC May 2009.

  > See the previous copyright statement (Springer-Verlag).

# An Empirical Investigation into the Security of Phone Features in SIP-based VoIP Systems

Ruishan Zhang[1], Xinyuan Wang[1], Xiaohui Yang[1], Ryan Farley[1], and Xuxian Jiang[2]

[1] George Mason University, Fairfax, VA 22030, USA
zhangruishan@gmail.edu, {xwangc, xyang3,rfarley3}@gmu.edu
[2] N.C. State University, Raleigh, NC 27606, USA
jiang@cs.ncsu.edu

**Abstract.** Phone features, e.g., *911 call*, *voicemail*, and *Do Not Disturb*, are critical and necessary for all deployed VoIP systems. In this paper, we empirically investigate the security of these phone features. We have implemented a number of attacks and experimented with VoIP services by leading VoIP service providers Vonage, AT&T and Gizmo. Our experimental results demonstrate that a man-in-the-middle or remote attacker could transparently 1) hijack selected E911 calls and impersonate the Public Safety Answering Point (PSAP); and 2) spoof the voicemail servers of both the caller and the callee of selected VoIP calls; and 3) make spam calls to VoIP subscribers even if *Do Not Disturb* is enabled. These empirical results confirm that leading deployed SIP-based VoIP systems have serious security vulnerabilities.

**Key words:** VoIP security, SIP, voicemail fraud, 911 hijacking, voice spam

## 1 Introduction

In addition to the basic function of making and receiving a call, VoIP systems generally offer many phone features, e.g., *911 call*, *voicemail*, and *Do Not Disturb*. Phone features are critical and necessary for all deployed Public Switched Telephone Network (PSTN) and VoIP systems.

Among all phone features, 911 emergency call is perhaps the most critical one. Recognizing that proper function of 911 call could impact the "life or death for millions of customers that subscribe to VoIP service" [1], the Federal Communications Commission (FCC) requires that all the interconnected VoIP services must support Enhanced 911 (E911) call and automatically report the caller ID and the registered location of E911 calls. On the other hand, voicemail is one of the most frequently used features of VoIP service. It is estimated [2] that 60∼70% of phone calls will be answered by voicemail rather than human. Given that a voice message often contains personal and sensitive information, any compromise of voicemail would violate the privacy of both the sender and the receiver of the voice message. *Do Not Disturb* allows VoIP users to temporarily block all

incoming phone calls and have some quiet time. When *Do Not Disturb* fails to work, spam phone calls might continually annoy VoIP users.

Signaling channel and voice channel are two most important components of VoIP systems. In current deployed systems, the Session Initiation Protocol (SIP) [3] and the Real Time Transport Protocol (RTP) [4] are the most dominant signaling and voice transport protocol, and being widely used.

Although the VoIP features are intuitively expected as trustworthy and reliable as those in the traditional PSTN, the open architecture of VoIP has enabled many attacks on voice communication that were not possible in the traditional PSTN. In this paper, we empirically investigate and evaluate the security of phone features in currently deployed SIP-based VoIP systems in the U.S., e.g., Vonage [5], AT&T's CallVantage [6] and Gizmo [7]. Specifically, we focus on the trustworthiness of *E911 call*, *voicemail*, and *Do Not Disturb*.

Assuming there exists a man-in-the-middle (MITM) in between the VoIP phones and the VoIP servers or a remote attacker on the Internet, we implement a number of MITM and remote attacks on the investigated voice services. Specifically, the MITM can transparently hijack selected 911 emergence calls and impersonate the PSAP. The MITM could also launch various voicemail fraud attacks against selected VoIP subscribers. When a VoIP phone makes a call to a PSTN phone or receives a call from a PSTN phone, the MITM can impersonate the callee's voicemail server and ask the caller to leave a voice message or call another phone number. In addition, the MITM can impersonate the voicemail server when the caller accesses voicemail. This would allow the attacker to generate arbitrary fake voice messages to the caller. Finally, a remote attacker,not necessarily a MITM, can circumvent *Do Not Disturb* and make arbitrary calls to Vonage and AT&T VoIP phones. Our experiments confirm that currently deployed VoIP systems are far from secure and trustworthy.

The rest of this paper is organized as follows. Section 2 briefly overviews SIP and SIP security mechanisms. Section 3 introduces our exploitation methodology. Section 4 describes our experiments on E911, voicemail and *Do Not Disturb*. Section 5 discusses the root causes of the vulnerabilities in deployed VoIP systems, and proposes some approaches to mitigate potential threats. Section 6 introduces related work. Finally, section 7 concludes the paper.

## 2 SIP overview

*Session Initiation Protocol* (SIP) [3], is a HTTP-like, application layer signaling protocol used for creating, modifying, and terminating multimedia sessions (e.g. VoIP calls) among Internet endpoints. SIP signaling involves various 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 acts on behalf of UA to forward the SIP messages to its destination.

The SIP specification [3] recommends using TLS or IPSec to protect SIP signaling messages. It also suggests using S/MIME to protect the integrity and

confidentiality of SIP message bodies. However, most deployed SIP VoIP systems only utilize SIP authentication to protect SIP messages. Fig. 1 shows the typical SIP authentication of call setup and termination.

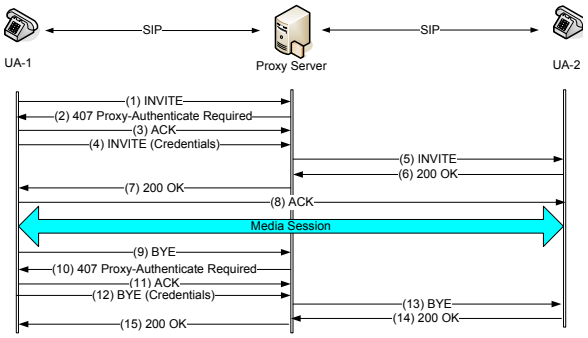


**Fig. 1.** An Example of Message Flow of SIP Authentication for INVITE and BYE Messages

The SIP authentication of currently deployed VoIP systems has the following weaknesses [3][8]:

- It only protects a few important SIP messages.
- It only protects a few SIP fields.
- It only authenticates SIP messages from SIP phones to the SIP servers.

## 3 Investigation Approach

Our approach of the security investigation is from the perspective of VoIP customers rather than VoIP service providers. Consequently, we leave aside the attacks on the VoIP service provider's servers and instead focus on those attacks that directly target the VoIP users. We choose to experiment with the residential VoIP services by Vonage, AT&T, who are the No.1 and No.2 [9] in U.S. VoIP market share. In addition, we experiment with Gizmo's popular softphone that has been used by millions of people. Note all the attacks we experiment are against our own accounts and phones only.

The key technique used in our empirical investigation is the MITM who can monitor, modify and forge VoIP traffic to or from selected VoIP users. Since most VoIP phones are many hops away from the VoIP servers, attackers have many opportunities to play MITM attack on existing deployed VoIP services. In fact, Zhang et al [10] have shown that a remote attacker from anywhere on the Internet, who is not originally in between the targeted VoIP phone and its VoIP servers, can become a MITM within a few minutes by exploiting some implementation flaws in the VoIP phone.

The MITM is used to verify the weaknesses in E911 and voicemail. To circumvent *Do Not Disturb*, the MITM is not required. A remote attacker can successfully make spam calls to defeat this feature.

## 4 Experiments with Deployed VoIP Services

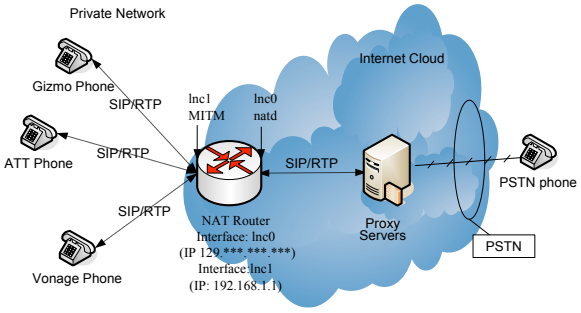


**Fig. 2.** Testbed Setup of MITM Attacks

To empirically investigate the security of deployed VoIP systems, we build a testbed that consists of the MITM, Vonage, AT&T SIP phones and Gizmo's softphone. Fig. 2 illustrates the network setup of our testbed. All the SIP phones are within a private network 192.168.1.x, where the Gizmo softphone runs on a Windows XP virtual machine. The MITM runs on a FreeBSD 5.4 virtual machine, which acts as the Network Address Translation (NAT) router for the private network. Specifically, the natd runs on the external interface `lnc0` of the FreeBSD virtual machine, and our MITM intercepts, manipulates the network traffic at the internal interface `lnc1` via divert socket. Note the MITM does not need to be directly connected to the VoIP phones, and it could be at anywhere along the path of VoIP traffic.

### 4.1 911 Call Hijacking

When a VoIP user dials 911, the call is supposed to be routed to a geographically appropriate PSAP by the VoIP service provider. However, our experiments show that the MITM could hijack the selected 911 call from either Vonage or AT&T SIP phone, divert it to any third party and let the third party impersonate the PSAP. In this case, the 911 call is never routed to the VoIP service provider, and yet it appears to the 911 caller that the 911 call is successfully connected to the appropriate PSAP.

Fig. 3 illustrates the message flows of the 911 call hijacking experiments. Specifically, the left and the right parts show the message flows of 911 calls from the Vonage phone and the AT&T phone respectively. Depending on the service

provider's implementation, the signaling path and the RTP stream path could be different. All SIP and RTP packets are transferred on UDP. We use SIP/RTP server(s) to denote the SIP server and the RTP server which handle the signaling messages and the RTP streams respectively.

When we dial 911 from the Vonage (or the AT&T) SIP phone[3], the caller's SIP phone sends an `INVITE sip:911` message to the SIP server in step (1) or (1'). Our MITM intercepts the `INVITE` message, pretends to be the SIP server and responds with a spoofed `200 OK` message in step (2) or (2'). In the spoofed `200 OK` message, the MITM sets its own IP address and port number (e.g., 12345) as the RTP stream termination point, which asks the caller's SIP phone to establish the voice stream to the MITM instead of the service provider's server. Although Vonage and AT&T's SIP server normally challenges the `INVITE` message with `407 proxy-authentication required` as shown in Fig. 1, we find that both Vonage and AT&T SIP phones actually accept the spoofed `200 OK` message directly, and they respond with an `ACK` message to the SIP server in step (3) or (3'). The MITM intercepts the `ACK` message so that it won't reach the service provider's SIP server. At this point, the three way handshake for the VoIP call setup is finished, and the 911 call between the caller and the MITM has been established. Then at step (4) or (4'), the caller talks to the MITM, who pretends to be the PSAP.

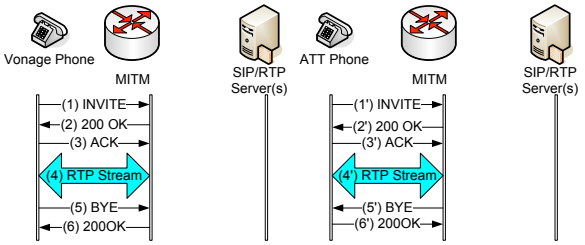


**Fig. 3.** Message Flow of Hijacking 911 Calls

Traditionally, 911 calls can only be terminated by the PSAP. We find that the Vonage SIP phone actually allows the 911 caller to terminate the 911 call. Specifically, the Vonage SIP phone sends out a `BYE` message to the Vonage's SIP server once the 911 caller hangs up in step (5). Then the MITM simply responds with a fake `200 OK` message. On the other hand, the AT&T SIP phone prevents the 911 caller from terminating the call until it is reset. Specifically, if the 911 caller hangs up the AT&T SIP phone, it starts and keeps ringing until the handset is picked up or the phone adapter is reset. This behavior conforms to the specification of traditional 911 call in the PSTN. At step (5'), the MITM pretends the PSAP and sends the AT&T SIP phone a fake `BYE` message. The

[3] Note we block our experimental 911 calls at our border router so that they will not interfere with any real 911 calls

AT&T SIP phone responds with a `200 OK` message which would terminate the 911 call completely.

Despite the 911 call implementation differences between Vonage and AT&T SIP phones, our MITM is able to transparently hijack selected 911 calls and pretend to be the PSAP. Our experiments demonstrate that 911 calls from existing deployed Vonage and AT&T VoIP services are not trustworthy.

### 4.2 Fake Voicemail Attacks

Voicemail is intended for the caller to leave a voice message when the callee is not available. Once the caller hears the voicemail prompt after dialing the callee's phone number, he would expect it is the callee's voicemail. In addition, when someone wants to check his voicemail and dials his voicemail access number, he would expect to reach his own voicemail. In this section, we show that the MITM can compromise the trust of voicemail by spoofing both the caller's voicemail and callee's voicemail.

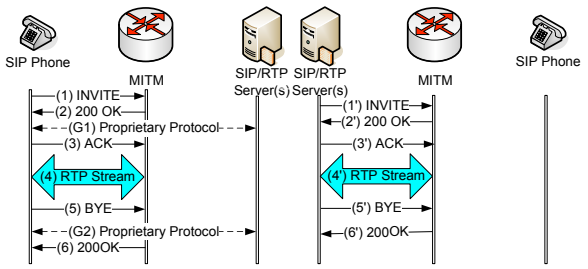


**Fig. 4.** Message Flows of Fake Callee's Voicemail at Caller's and Callee's Sides

**Fake Callee's Voicemail Attack** When the caller places a call, the MITM intercepts the call and spoofs the callee's voicemail to prompt the caller to leave a voice message or call some other phone number. In this case, the caller is tricked into believing that the callee is not available even if the callee is actually available. Note the fake callee's voicemail attack can be launched by the MITM at either the caller's or callee's side. Therefore, the fake callee's voicemail attack is possible even if one communication party (caller or callee) uses a PSTN phone.

**Fake callee's voicemail at the caller's side** We use our Vonage and AT&T SIP phones to call a PSTN phone. The left part of Fig. 4 illustrates the corresponding message flow. After dialing the callee's phone number, the caller's SIP phone sends the corresponding `INVITE` message to the SIP server of its service provider. The MITM intercepts the `INVITE` message in step (1), and replies with a fake `200 OK` message in step (2). The caller's SIP phone is tricked by the fake `200 OK` message and sends back an `ACK` message in step (3). This

establishes the voice RTP session between the caller's SIP phone and the MITM. In step (4), the MITM sends the crafted RTP stream to the caller's SIP phone and the caller hears bogus voice greeting "xxx can not take your call, please leave a brief message or call 1-xxx-xxx-xxx". Then the MITM starts to record the RTP stream from the caller's SIP phone. Once the caller hangs up, his SIP phone sends a `BYE` message to the MITM in step (5), the MITM replies with a `200 OK` message to the caller's SIP phone in step (6), which terminates the call.

When we use the Gizmo softphone to call the PSTN phone, the Gizmo softphone does not respond with an `ACK` message to the MITM after receiving the `200 OK` message from the MITM in step (2). After further investigation, we find that the Gizmo phone does not follow the SIP specification [3] exactly:

- The IP address and the port number of the RTP server are specified by the Gizmo softphone in the `INVITE` message in step (1), rather than by the SIP server in the `200 OK` message in step (3).
- Gizmo softphone uses some proprietary protocol, running on TCP port 443, to exchange some signaling information with Gizmo RTP server in step (G1) and (G2). Step (G1) is necessary for establishing the SIP call, and if packets in step (G1) are dropped by the MITM, the Gizmo softphone will not proceed to step (3). Once the MITM allows the traffic in step (G1), the the Gizmo softphone proceeds to establish the call with its SIP server. Step (G2) is used to terminate the SIP call.

After implementing these special handling, the MITM is able to spoof the callee's voicemail to calls from the Gizmo softphone.

**Fake callee's voicemail at the callee's side** In this experiment, we use a PSTN phone to call our Vonage and AT&T SIP phones. The right part of Fig. 4 illustrates the corresponding message flow. After the caller dials the phone number of a SIP phone from a PSTN phone, the SIP server sends an `INVITE` message to the SIP phone in step (1'). The MITM intercept the `INVITE` message and responds with a `200 OK` message in step (2'). The SIP server thinks it is from the SIP phone and sends an `ACK` message in step (3'). Now the voice RTP session between the SIP server and the MITM is established, and the MITM sends the RTP server crafted RTP stream in step (4'). As a result, the caller from the PSTN phone hears the bogus voice greeting "XXX can not take your call, please leave a brief message or call 1-xxx-xxx-xxx". Then the MITM starts to record the RTP stream from the RTP server. Once the caller hangs up, the SIP server sends a `BYE` message to the MITM in step (5'). Finally, the MITM replies with a `200 OK` message to the SIP server in step (6'), which terminates the call.

**Fake Caller's Voicemail Attack** When a caller wants to check his voicemail, he usually dials some voicemail access number (e.g., *123 for Vonage, *** for AT&T, 611 for Gizmo) and authenticates himself with a voicemail PIN. Again, the MITM can intercept the call and spoof the caller's voicemail. This not only allows the attacker to trick the caller with bogus voicemail messages but also let the attacker capture the caller's voicemail PIN.

**Table 1.** Differences between the Voicemail Services by Vonage, AT&T and Gizmo

| Service Provider | Voicemail Access Number | Codec for voice | Payload Type for Event |
|---|---|---|---|
| Vonage | *123 | G.711 PCMU(0) | 101 |
| AT&T | *** | G.721(2) | 100 |
| Gizmo | 611 | iLBC(102) | 106 |

We have experimented the fake caller's voicemail attack with Vonage, AT&T and Gizmo SIP phones. The SIP message flow is similar to the left part of Fig. 4. The MITM first spoofs the SIP server and establishes a RTP voice session with the voicemail caller. Then the MITM spoofs the voicemail service and interacts with the voicemail caller. In our implementation, we record the RTP stream from the real voicemail server and replay the recorded RTP stream to the voicemail caller. Therefore, what the voicemail caller hears is exactly the same as that from the real voicemail server. After the MITM prompts the voicemail caller, it waits for responses (e.g., PIN, functional choice) from the caller and responds accordingly.

We find that Vonage, AT&T and Gizmo use different codecs and RTP payload types for their voicemail traffic. Table 1 summarizes their differences in the codec and RTP payload type used. Specifically, the codecs for the RTP voice are G.711 PCMU, G.721 and iLBC respectively, and the payload types for the RTP event are 101, 100 and 106 respectively. Despite these differences in the voicemail implementation by Vonage, AT&T and Gizmo, our MITM is able to spoof the voicemail for all of them.

In summary, the MITM can spoof both the caller's and the callee's voicemail even if one side of the voice call uses a PSTN phone. Eventually, the caller is deceived to leave a voicemail or call another number while the callee does not even know he has been called. Meanwhile, the MITM can read the voicemail left by the caller and trick the voicemail caller with bogus voicemail messages. Furthermore, the MITM could obtain the voicemail PIN entered by the caller.

### 4.3 Circumventing Do Not Disturb

*Do Not Disturb* enables VoIP users to block all incoming phone calls during a short time, e.g., 30 minutes. When we used a PSTN phone to call a Vonage or an AT&T VoIP phone with Do Not Disturb enabled, the call was forwarded to the voicemail system. This indicates *Do Not Disturb* is effective if the call goes through SIP servers. However, we can circumvent *Do Not Disturb* by calling the Vonage or the AT&T phone directly.

Fig. 5 shows the network setup of circumventing *Do Not Disturb*. Note unlike previous experiment, this experiment only requires a remote attacker. Fig. 6 depicts the message flow of circumventing *Do Not Disturb* attacks on the Vonage phone and AT&T phone.

In step (1) and (1'), the remote attacker sends an `INVITE` message to the SIP phone. Note to make the `INVITE` message accepted by the SIP phone, the

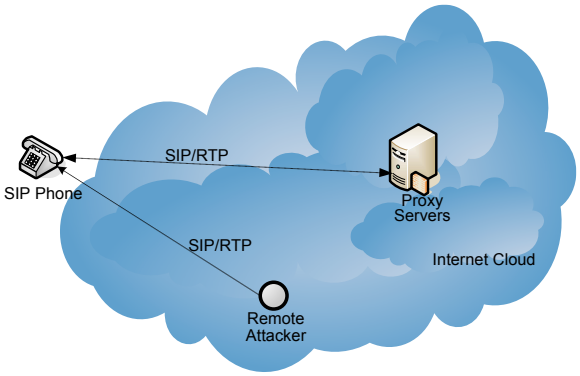


**Fig. 5.** Testbed Setup of Circumventing Do Not Disturb

IP address should be spoofed as that of a **real SIP server**. Otherwise, the SIP phone would omit this `INVITE` message. In the `INVITE` message, the IP address and port number for the RTP stream on the caller side are set to the remote attacker's IP address and 12345 respectively. Since the SIP specification does not require to authenticate SIP messages from SIP servers, the SIP phone will accept this `INVITE` message. Then in step (2-3) or (2'-3'), the SIP phone sends back `Trying,Ringing` messages to the real SIP server, and begins to ring. At this point, the called party begins to hear annoying ring tone. Consequently, *Do Not Disturb* is bypassed.

Actually, the attacker can even successfully establish a SIP call with these phones and send voice spam by exploiting some implementation flaws. After the receiver picks up the phone, the SIP phone responds with a `200 OK` message to its real SIP server. According to the SIP specification, to establish a SIP call, the

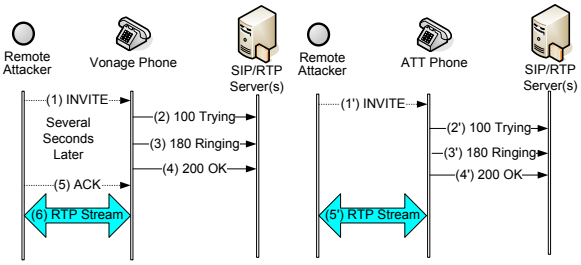


**Fig. 6.** Message Flow of Circumventing Do Not Disturb

remote attacker needs to send an `ACK` message to complete an `INVITE/200OK/ACK` three-way handshake. Since the `200 OK` message is sent to the real SIP server, the remote attacker does not know the time when the receiver picks up the phone. To solve this problem, The remote attacker can guess the time interval

between the `INVITE` message and the `200 OK` message, e.g., 5 seconds. Then 5 seconds after sending the `INVITE` message, the remote attacker sends an `ACK` message with a spoofed source IP address to the SIP phone. In addition, to ensure the correctness, the callee should check whether the Tag value in the To field of the `ACK` message is the exactly same as that in the `200 OK` message. Since the remote attacker can not see the `200 OK` sent to the SIP server, the remote attacker can not craft an `ACK` SIP message with an correct Tag value. After further investigation, we find:

- The Vonage phone does not check the Tag value, and accepts an `ACK` message with a random Tag value.
- The AT&T phone does not need a complete `INVITE/200OK/ACK` three-way handshake before sending RTP stream. As soon as the receiver picks up the phone, the AT&T phone begins to send RTP stream to the remote attacker.

Consequently, the remote attacker can exploit these implementation flaws to talk to both the Vonage and AT&T phone.

## 5 Discussion

We have demonstrated that a MITM or remote attacker, could successfully launch many attacks on phone features in deployed SIP-based VoIP systems. These attacks exploit the inherent vulnerabilities in current deployed VoIP systems.

- Only SIP authentication mechanism is employed to protect SIP messages. Due to the weaknesses in SIP authentication, an attacker can freely craft unprotected SIP messages (e.g., Trying, Ringing, 200 OK, ACK), and SIP messages from a SIP server to a SIP phone. In addition, the attacker can also modify unprotected SIP fields (e.g., SIP message body, From, To).
- Since RTP voice stream is unencrypted and unauthenticated, the attack can easily capture voice traffic, generate bogus RTP stream and talk to VoIP phones.

Leveraging these weaknesses in SIP and RTP, the attacker can seamless spoof the SIP and RTP server when talking to the caller, and seamless spoof the callee when talking to the SIP server and RTP server. Accordingly, the attacker can readily hijack 911 calls, spoof the caller's and the callee's voicemail, and make a call to the selected phone number. Note during an attack, both a VoIP phone and the PSTN or cell phone communicating with the VoIP phone are potential victims. In addition to the features that we have investigated in this paper, we believe other phone features, e.g., 411, all have similar exploitable vulnerabilities.

Since our attacks target VoIP end users, intrusion defense measures deployed on VoIP servers side are ineffective to mitigate the attacks. To defeat or minimize these attacks, the best solution is to provide privacy, integrity and authentication protection for SIP signaling messages and RTP voice streams. For example, SIP over TLS and Secure Real-time Transport Protocol (SRTP) could be deployed

to protect SIP messages and RTP voice streams. Unfortunately, we still have not widely seen such a deployment. Considering Vonage and AT&T Callvangtage's VoIP phones are being utilized to replace traditional PSTN phones, and millions of VoIP subscribers are using their VoIP phones to perform many security critical activities, e.g, phone banking and 911 calls, the current state of VoIP security is not optimistic.

## 6 Related Work

Most previous work is on the defense side. Arkko et al [11] proposed a scheme to negotiate the security mechanism used between a SIP phone and and its next-hop SIP entity. Baugher et al [12] proposed SRTP to protect the RTP traffic. However, none of them are widely being used. Reynolds et al [13] proposed multi-protocol protection against flood-based DoS attacks on VoIP networks. Wu et al [14] described a cross protocol intrusion detection architecture for VoIP environments. Sengar et al [15] proposed an intrusion detection system based on interactive protocol state machines. The above intrusion detection systems or methods are deployed on VoIP servers side, and ineffective to defend against the attacks we proposed.

Mintz-Habib et al [16] proposed a VoIP emergence service architecture and developed a prototype system. Zhang et al [8] [10] implemented four billing attacks on deployed VoIP systems, and demonstrated a remote attacker can become a MITM by exploiting some implementation flaws in a VoIP phone. Wang et al [17] systematically investigated the trust issue of SIP-based VoIP and identified the voice pharming attack on VoIP systems. McGann and Sicker [18] analyzed detection capability of several VoIP security tools: SiVuS,PROTOCOS [19], SIP Forum Test Framework [20], and some commercial products. They showed that there exists a large gap between known VoIP security vulnerabilities and the tool's detection capability.

## 7 Conclusion

In this paper, we empirically investigated the trustworthiness of phone features in leading deployed SIP-based VoIP systems. We demonstrated that the MITM could transparently hijack E911 calls and spoof the PSAP. In addition, it can spoof voicemail and use it for many potential voicemail related frauds. Finally, a remote attacker can circumvent *Do Not Disturb* and make annoying phone calls.

We hope our work can raise the awareness of millions of VoIP subscribers that the currently deployed VoIP phone features are not as trustworthy and reliable as expected. Before VoIP service providers employ SIP over SSL and SRTP to protect SIP signaling messages and RTP voice streams, a VoIP subscriber should be aware of the risk associated with current VoIP services.

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

## References


1. First Report and Order and Notice of Proposed RuleMaking, http://hraunfoss.fcc.gov/edocs_public/attachmatch/FCC-05-116A1.pdf
2. How To Deal With Voicemail When Prospecting, http://www.content4reprint.com/business/network-marketing/how-to-deal-with-voicemail-when-prospecting.htm
3. Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., Schooler, E.: SIP: Session Initiation Protocol. RFC 3261, IETF, June 2002.
4. Schulzrinne, H., Casner, S., Frederick, R., Jacobson, V.: RTP: A Transport Protocol for Real-Time Applications. RFC 1889, IETF, January 1996.
5. Vonage, http://www.vonage.com/
6. AT&T's CallVantage, https://www.callvantage.att.com/
7. Gizmo, http://www.gizmoproject.com/
8. Zhang, R., Wang, X., Yang, X., Jiang, X.: Billing Attacks on SIP-Based VoIP Systems. In: lst USENIX Workshop on Offensive Technologies (WOOT 2007), August 2007.
9. US VOIP market shares, http://blogs.zdnet.com/ITFacts/?p=11425
10. Zhang, R., Wang, X., Farley, R., Yang, X., Jiang, X.: On the Feasibility of Launching the Man-In-The-Middle Attacks on VoIP from Remote Attackers. In: 4th ACM Symposium on Information, Computer and Communications Security (ASIACCS 2009), Sydney, Australia, March 2009.
11. Arkko, J., Torvinen, V., Camarillo, G., Niemi, A., Haukka, T.: Security Mechanism Agreement for the Session Initiation Protocol (SIP). RFC 3329, IETF, January 2003.
12. Baugher, M., McGrew, D., Naslund, M., Carrara, E., Norrman, K.: The Secure Real-time Transport Protocol (SRTP). RFC 3711, IETF, March 2004.
13. Reynolds, B., Ghosal, D.: Secure IP Telephony Using Multi-layered Protection. In: 10th Network and Distributed System Security Symposium (NDSS 2003), Feburary 2003.
14. Wu, Y., Bagchi, S., Garg, S., Singh, N.: SCIDIVE: A Stateful and Cross Protocol Intrusion Detection Architecture for Voice-over-IP Environments. In: 34th International Conference on Dependable Systems and Networks (DSN 2004), Pages 433 – 442, July 2004.
15. Sengar, H., Wijesekera, D. Wang, H., Jajodia, S.: VoIP Intrusion Detection Through Interacting Protocol State Machines. In: 36th International Conference on Dependable Systems and Networks (DSN 2006), June 2006.
16. Mintz-Habib, M., Rawat, A., Schulzrinne, H., Wu, X.: A VoIP Emergency Services Architecture and Prototype. In: 14th International Conference on Computer Communications and Networks (ICCCN 2005), October 2005.
17. Wang, X., Zhang, R., Yang, X., Jiang, X., Wijesekera, D.: Voice Pharming Attack and the Trust of VoIP. In: 4th International Conference on Security and Privacy in Communication Networks (SecureComm 2008), September 2008.
18. McGann, S., Sicker, D. C. : An analysis of Security Threats and Tools in SIP-Based VoIP Systems. In: Second VoIP Security Workshop, 2005.
19. PROTOS SIP Fuzzer, http://www.ee.oulu.fi/research/ouspg/protos/testing/c07/sip/
20. SIP Forum Test Framework, http://www.sipfoundry.org/sip-forum-test-framework/sip-forum-test-framework-sftf.html