Billing Attacks on SIP-Based VoIP Systems

Ruishan Zhang, Xinyuan Wang, Xiaohui Yang, Xuxian Jiang
Department of Information and Software Engineering
George Mason University, Fairfax, VA 22030, USA
{*rzhang3, xwangc, xyang3, xjiang*}*@gmu.edu*

Abstract

Billing is fundamental to any commercial VoIP services and it has direct impact on each individual VoIP subscriber. One of the most basic requirements of any VoIP billing function is that it must be reliable and trustworthy. From the VoIP subscriber's perspective, VoIP billing should only charge them for the calls they have really made and for the duration they have called.

Existing VoIP billing is based on VoIP signaling. Therefore, any vulnerability in VoIP signaling is a potential vulnerability of VoIP billing. In this paper, we examine how the vulnerabilities of SIP can be exploited to compromise the reliability and trustworthiness of the billing of SIP-based VoIP systems. Specifically, we focus on the billing attacks that will create inconsistencies between what the VoIP subscribers received and what the VoIP service providers have provided. We present four billing attacks on VoIP subscribers that could result in charges on the calls the subscribers have not made or overcharges on the VoIP calls the subscribers have made. Our experiments show that Vonage and AT&T VoIP subscribers are vulnerable to these billing attacks.

1 Introduction

VoIP is becoming increasingly popular due to its advantages in cost and functionality. IDC [5] predicted that the number of US residential VoIP subscribers will grow from 10.3 million in 2006 to 44 million by 2010.

Billing is one of the most fundamental component of any commercial VoIP services and it has direct impact on each individual VoIP subscriber. One basic requirement of any VoIP billing function is that it must be reliable and trustworthy. For example, VoIP service providers depend on billing to charge their customers for all the billable services and they do not want to lose any revenues from any billable services they provide. On the other hand, VoIP subscribers expect the billing be accurate so that they will be charged only for the calls they have made and for the duration they have really called. In addition, the VoIP billing should be resilient to billing fraud and be free of any inconsistency between what the service providers have provided and what the customers have received.

Existing VoIP billing is based on VoIP signaling. Session Initiation Protocol (SIP) is the dominant VoIP signaling protocol, and it is being used widely in commercial VoIP services. Therefore, any vulnerability in SIP could make the billing of many commercial SIP-based VoIP systems vulnerable.

In this paper, we examine how the vulnerabilities of SIP can be exploited to compromise the reliability and trustworthiness of the billing of SIP-based VoIP systems. Specifically, we focus on the billing attacks that will create inconsistencies between the what the VoIP subscribers have received and what the VoIP service providers have provided. We present four billing attacks on VoIP subscribers 1) *InviteReplay*; 2) *FakeBusy*; 3) *ByeDelay*; and 4) *ByeDrop* that could either make calls without subscriber's authorization or prolong the duration of subscriber's call transparently. For calls (e.g. international call) that are charged with a per minute rate, these billing attacks will result in either charges on the calls the subscribers have not made or overcharges on the VoIP calls the subscribers have made. Note these VoIP billing attacks do not require any collaborations from the SIP servers or SIP phones and they could launched without knowledge of the secret password shared between the SIP server and SIP phone. While the FakeBusy, ByeDelay and ByeDrop billing attacks need the *man in the middle* (MITM) at both ends of the signaling path, the InviteReplay billing attack can be launched from virtually anywhere once the attacker has obtained a copy of one legitimate `INVITE` message from the victim's SIP phone.

We have implemented these billing attacks against VoIP subscribers and experimented with our VoIP accounts of Vonage [1] and AT&T [9]. Our experiments

show that AT&T VoIP subscribers are vulnerable to all the above mentioned four billing attacks, and Vonage VoIP subscribers are vulnerable to the FakeBusy, ByeDelay and ByeDrop attacks. In either case, the VoIP subscribers could be overcharged due to the billing attacks. Since these billing overcharges are not caused by the errors of the billing system itself, but rather the exploits of the vulnerabilities of SIP, they will create hard-to-resolve disputes between the VoIP subscribers and the service providers. One immediate consequence of the identification of these billing attacks is that the billing of existing VoIP services becomes questionable and untrustworthy.

The rest of this paper is organized as follows. Section 2 gives an overview of SIP and its security mechanism. Section 3 presents four billing attacks on SIP-based VoIP subscribers. Section 4 describes related works. Section 5 concludes the paper.

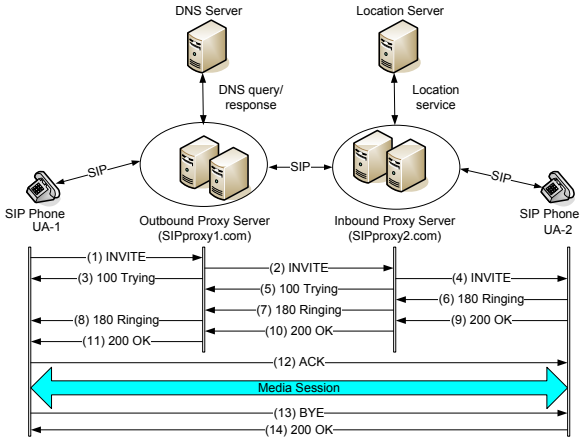

Figure 1: An Example of SIP Flow of Call Setup and Tear Down

2 SIP Overview

Session Initiation Protocol (SIP) [19], is a general purpose, application layer signaling protocol used for creating, modifying, and terminating multimedia sessions (e.g. VoIP calls) among Internet endpoints. SIP defines the signaling interaction between: *user agent* (UA), *proxy server*, *redirect server*, *registrar server* and *location server*. An UA represents an endpoint of the communication (i.e., a SIP phone). Based on its role in the communication, an UA could be either UA client or UA server. The proxy server is the intermediate server that acts on behalf of UA to forward the SIP messages to its destination. The registrar server handles the UA's registration request. The location server maintains the location information of the registered UAs. The redirect server provides the UA client with an alternative set of contact addresses on behalf of the UA server.

SIP is based on an HTTP-like request/response model. To set up, manage or terminate a VoIP session, UA client (UAC) sends a SIP request message to a SIP server or UA server (UAS). Then the SIP server or UAS replies with a SIP response message identified by a status code that indicates the outcome of the request. Each user in SIP network is identified by a SIP *Uniform Resource Identifier* (URI), which usually contains a username and hostname. Figure 1 shows a typical SIP message flow of a call setup and tear down. When the caller (UA-1) begins to initiate a call to the callee (UA-2), it sends an `INVITE` message to its outbound proxy server at domain SIPproxy1.com. Upon receiving the `INVITE` message, the outbound server locates the inbound server at domain SIPproxy2.com via Domain Name Service (DNS), and forwards the `INVITE` message to the inbound server. Meanwhile, the outbound proxy server sends back a `100 TRYING` message to UA-1 which means that outbound proxy has received the request and it is working on forwarding the `INVITE` message to its destination. After receiving the `INVITE` message, the inbound proxy server gets the current location (i.e. IP address) of UA-2 by querying the location service, then it forwards the `INVITE` message to UA-2. Upon receiving the `INVITE` message, UA-2 rings and replies with a `180 Ringing` message to UA-1 so that the caller can hear the ringback tone. When the callee picks up the phone, UA-2 sends back a `200 OK` message to UA-1 to inform that the call has been answered. Upon receiving the `200 OK` message, UA-1 stops the ringback tone and sends back an `ACK` message to UA-2. After UA-2 receives the `ACK` message, the three way handshake is completed and the VoIP session is established. Note the message bodies of the `INVITE` message and `200 OK` message contain the negotiated media session parameters (e.g., codec, IP address and port number of the RTP stream) specified in Session Description Protocol (SDP) [10]. Now UA-1 and UA-2 begins to send RTP [21] voice streams to each other based on the negotiated media session parameters. At the end of the call, UA-1 (UA-2) hangs up first and sends a `BYE` message to its peer. After receiving the `BYE` message, UA-2 (UA-1) sends back a `200 OK` message and stops sending its RTP stream to its peer. Upon receiving the `200 OK`, UA-1 (UA-2) stops sending its RTP streams to its peer. Then the SIP session is terminated.

2.1 SIP Security

The SIP security is largely based on existing security mechanisms for HTTP and SMTP. The SIP specification [19] recommends using TLS [6] or IPSec [11] to protect the SIP signaling path in SIP networks. It suggests using S/MIME[14] to protect the integrity and confiden-

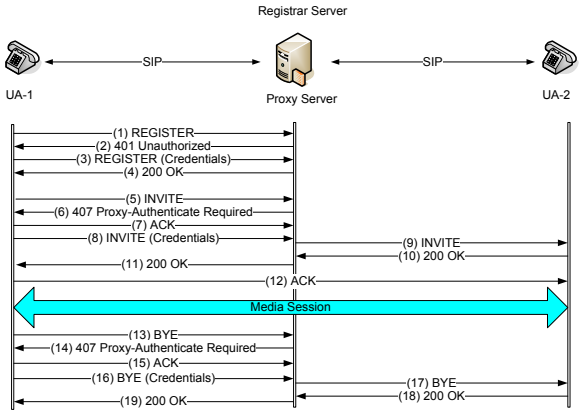

Figure 2: An Example of Message Flow of SIP Authentication for REGISTER, INVITE and BYE Messages

tiality of SIP messages. However, it is difficult to protect the SIP message from end-to-end since intermediate SIP servers need to examine and change certain fields of the SIP messages while they are transferred. SIP mandates that all SIP proxies, redirect servers and registration servers must support TLS [6] and HTTP digest based authentication [7]. However, UAs are required to support HTTP digest based authentication [7] only.

Based on HTTP digest authentication [7], SIP authentication provides anti-replay protection and one-way authentication to SIP messages. It can be used by a SIP UAC, SIP UAS, SIP proxy or registrar server to prove that it knows the shared secret password. Figure 2 shows the typical SIP authentication of call registration, call setup and termination. When a SIP server (e.g. proxy, registrar) receives a SIP request (e.g. `INVITE`, `REGISTER`, `BYE`), the SIP server challenges the UAC with either a `401 unauthorized` or a `407 proxy-authencation required` message. Upon receiving the `401` or `407` response, the UAC applies specific digest algorithm (e.g., MD5 [17]) to SIP message fields *request-URI*, *username*, *password*, *realm*, and *nonce* to get a hash value. Then the UAC resend the SIP request with the hash value as part of the credential to authenticate the SIP request.

However, existing SIP authentication has the following weaknesses:

- It only applies to a few SIP messages (e.g., `INVITE`, `BYE`, `REGISTER`), and it leaves other important SIP messages (e.g., `TRYING`, `RINGING`, `200 OK`, `ACK` and `BUSY`) unprotected.

- It only protects a few SIP fields (e.g., `request-URI`, `username`, `realm`), and it leaves other important SIP fields (e.g., `SDP`, `From`, `To`) unprotected.

- It only applies to SIP messages from the UAC (i.e., SIP phone) to SIP servers, and it leaves all the SIP messages from the SIP servers to UAC unprotected.

Since UAs are not required to support any link level encryption (e.g., TLS, IPsec), the SIP messages between the SIP servers and the UAs are in clear text. Therefore, any *man-in-the-middle* (MITM) in between the SIP server and the SIP UA can freely modify those fields that are not protected by the SIP authentication. Furthermore, the MITM can freely spoof any SIP messages from any SIP server to any particular SIP UA since the SIP messages from SIP servers to SIP UAs are not authenticated at all. All these vulnerabilities in SIP authentication make it possible to manipulate the SIP messages to corrupt the billing of SIP-based VoIP systems.

3 Billing Attacks on SIP-based VoIP Systems

In this section, we discuss how the vulnerabilities in SIP can be exploited to launch billing attacks against SIP-based VoIP systems. Specifically, we focus on those billing attacks that target subscribers of SIP-based VoIP systems.

Existing commercial VoIP services may have either *unlimited* or *limited* call times of certain calls (e.g., domestic or international call to selected countries). If the VoIP subscribers call some numbers (e.g., international or 900) that are not covered in the unlimited plan, they have to pay those calls on a per minute basis. In addition, the VoIP subscriber of a limited service plan (e.g., 500 minutes/month) needs to pay any calls that are over the call time limit. In these cases, if the attacker could somehow make unauthorized calls or prolong the duration of calls made by the VoIP subscribers, the attacker can make the selected VoIP subscriber to pay more than he/she should.

One key component in all our billing attacks is the use of MITM. Given that VoIP service providers usually operate one or a few SIP servers for call setup, most SIP phones will be hundreds or even thousands of miles away from the SIP signaling server. This would give the attacker many opportunities to play the MITM in the public Internet.

We describe four such billing attacks against subscribers of leading VoIP service providers (e.g., Vonage, AT&T): 1)InviteReplay billing attack; 2) FakBusy billing attack; 3) ByeDelay billing attack; and 4) ByeDrop billing attack. Note all the billing attacks we experimented were against ourselves rather than any other VoIP subscribers. At no time did we send any VoIP traffic to negatively affect the VoIP infrastructure or violate any service agreement.

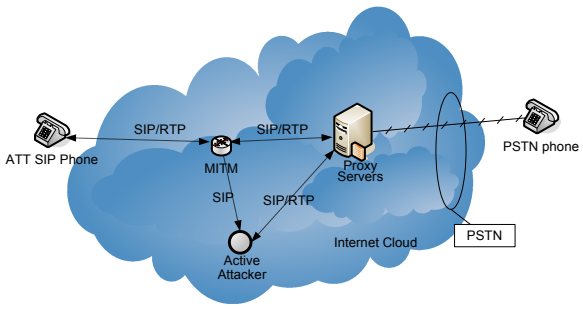

Figure 3: InviteReplay Billing Attack against AT&T SIP Phone

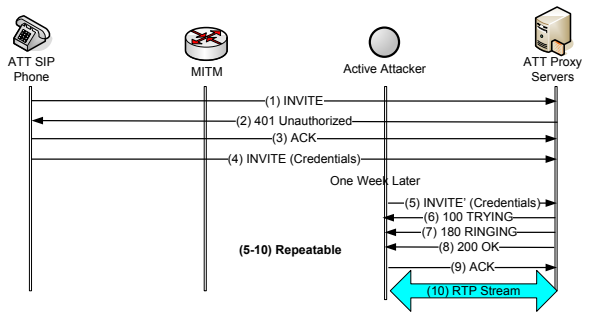

Figure 4: Message Flow of InviteReplay Billing Attack against AT&T SIP Phone

3.1 InviteReplay Billing Attack

InviteReplay billing attack aims to make unauthorized calls by replaying intercepted `INVITE` messages. Such a billing attack exploits the implementation errors of the anti-replay functionality in SIP authentication, and it could be effective even if the `INVITE` messages are protected by SIP authentication.

Figure 3 illustrates the big picture of InviteReplay billing attack against AT&T SIP VoIP subscribers. The MITM who is in between the AT&T SIP phone and the AT&T SIP server can observe and intercept all the SIP messages sent by the AT&T SIP phone. The MITM can send the intercepted `INVITE` message to another attacker, who can make unauthorized SIP calls by replaying the modified `INVITE` message. Figure 4 shows the message flow of the InviteReplay billing attack. Steps (1-4) show that when the AT&T SIP phone (xxx-xxx-0451) calls a PSTN phone (xxx-xxx-9398), the SIP phone need to authenticates the `INVITE` message to the SIP proxy upon request. The MITM can eavesdrop the INVITE message with the authentication credentials, and send it to the remote active attacker. The remote attacker can freely modify the RTP session parameters (e.g., IP address and port number) specified in the SDP part of the INVITE message since they are not protected by the SIP authentication. Then the attacker can repeatedly mount InviteReplay billing attack by replaying the modified `INVITE` message as shown in steps (5-10). After step (5-9), the call is established between the attacker and the SIP server. In step 10, the attacker and the SIP proxy exchange RTP streams to each other according to negotiated session parameters (e.g., IP address and port number) specified in the `INVITE` and `200 OK` messages. Now the active attacker could either speak to the callee or play some recorded voice message. After step (5-10), the online activity system of AT&T's CallVantage shows that AT&T SIP phone made another call to the PSTN phone although the original AT&T VoIP subscriber did not call the callee again.

We measure the call duration by collecting network traffic. Interestingly, the RTP streams of every unauthorized call last about 3 minutes. After 3 minutes, the AT&T SIP proxy sends the attacker an `INVITE` *sip:******0451@192.168.1.118:5060 SIP/2.0* message and stops sending RTP streams. We suspect that this is due to the periodic registration by the AT&T SIP phone, which would allow AT&T SIP proxy to identify the difference between the current location (i.e., IP address) of the AT&T SIP phone and the IP address to where it is sending the RTP stream. However, since the IP address of the `REGISTER` message is not protected by the SIP authentication, it can be easily spoofed as well. Therefore, it is possible to make the unauthorized VoIP calls longer than 3 minutes.

Our experiences show that the intercepted `INVITE` message can be replayed successfully one week after it has been intercepted. This means that the attacker could repeatedly launch the InviteReplay billing attack against the targeted AT&T VoIP subscriber.

Our experiments with our Vonage account show that Vonage subscribers are immune from such InviteReplay billing attack. This suggests that Vonage SIP server has implemented the anti-replay function correctly.

3.2 FakeBusy Billing Attack

FakeBusy billing attack essentially hijacks VoIP calls of targeted VoIP subscriber and controls the VoIP call duration. As a result, the call attempted by the VoIP subscriber would fail, and yet the VoIP subscriber will be billed for the call of duration determined by the attacker.

Figure 5 illustrates the network setup of FakeBusy billing attack, ByeDelay billing attack and ByeDrop billing attack. There exist two MITMs: MITM1 stands between the SIP phone (e.g., Vonage phone) and the SIP servers (e.g., Vonage proxy servers) at one end, and MITM2 stands between the SIP phone (e.g., AT&T phone) and the SIP servers (e.g., AT&T proxy servers) of the other end.

Figure 6 shows the message flow of the FakeBusy

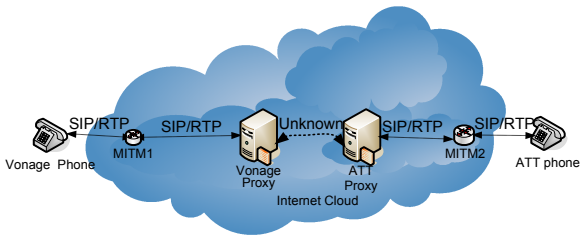

Figure 5: Network Setup of FakeBusy-billing Attack, ByeDelay-billing Attack and ByeDrop-billing Attack

billing attack when a Vonage phone calls an AT&T phone. The left and right parts show the message flows of the caller side and the callee side, respectively. Note that the signaling path and the RTP streams path are not necessarily the same. We use SIP server to denote the server (proxy) that handles signaling messages, and use RTP server to denote the server that handles the RTP streams. In step (1-4), the caller authenticates the `INVITE` message to the Vonage SIP server. In step 4, MITM1 intercepts the `INVITE` message with authentication credentials and modifies the IP address and the port number for RTP stream to its own IP and a chosen port number (e.g., 22222). In step 5, MITM1 sends the modified `INVITE` message to the Vonage SIP server. Upon receiving the modified `INVITE` message, the Vonage SIP server informs the AT&T SIP server that the Vonage phone wants to call the AT&T phone. Meanwhile, MITM1 sends a `BUSY` message to the caller, which will make the caller think that the callee is on the phone. In step 1', the AT&T SIP server sends an `INVITE` message to the callee. MITM2 intercepts the `INVITE` message and replies with `TRYING`, `RINGING` and `200 OK` messages. MITM2 specifies his own IP address and a chosen port number (e.g., 22222) in the `200 OK` message. This would let the AT&T servers send RTP stream to MITM2 rather than AT&T phone. The Vonage SIP server then sends `TRYING`, `RINGING`, `200 OK` message to MITM1. Now the IP address and the port number in the `200 OK` message point to the Vonage RTP server. MITM1 replies with an `ACK` message to the Vonage SIP server. Accordingly, the AT&T SIP server sends an `ACK` message to MITM2. Now the call has been successfully setup between MITM1 and MITM2, and the VoIP service providers (e.g., Vonage, AT&T) starts to count the time of the call.

In our experiments, we let the MITM1 and MITM2 exchange RTP streams for about 34 minutes before we let MITM2 terminate the call. To terminate the established call, MITM2 generates a `BYE` message and sends it to the AT&T SIP server. Since the AT&T SIP server doesn't require the `BYE` message to be authenticated, it will accept it, reply with a `200 OK` message and will ask the Vonage SIP servers to tear down the established VoIP call. Vonage's online call activity system shows that Vonage phone made a call of 34 minutes long to AT&T phone, while the caller thinks the attempted call has failed and the callee does not even know he has ever been called.

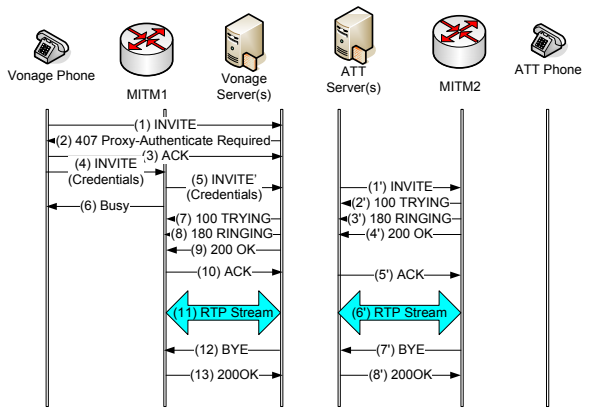

Figure 6: Message Flow of FakeBusy Billing Attack

3.3 ByeDelay Billing Attack

ByeDelay billing attack seeks to transparently prolong the duration of established calls between targeted VoIP subscribers by delaying the `BYE` messages. Figure 7 shows the message flow of the attack when a Vonage phone calls an AT&T phone. Steps (1-9) and (1'-6') are the same as the normal call. When the caller or the callee hangs up and sends a `BYE` message to its SIP server, MITMs intercept the `BYE` message and send back a `200 OK` message. This would give the caller or callee the impression that the call has successfully terminated while the MITMs have taken over the established call. In step 12 and 9', MITM1 and MITM2 generate bogus RTP streams and send them to Vonage and AT&T's RTP servers respectively. This would give the service providers the impression that the caller and the callee are still actively talking, and thus prolong the call duration to be billed.

In our experiments, we let the MITMs exchanging bogus RTP streams for about 19 minutes before letting MITM2 generate a `BYE` message and send it to the AT&T SIP server to terminate the prolonged called. Alternatively, we can let MITMs send the intercepted `BYE` messages. Vonage's online activity system showed that the call between the Vonage phone and the AT&T phone was 19 minutes longer than it actually was. Therefore, ByeDelay attack could effectively cause overcharges of the calls made by the VoIP subscribers.

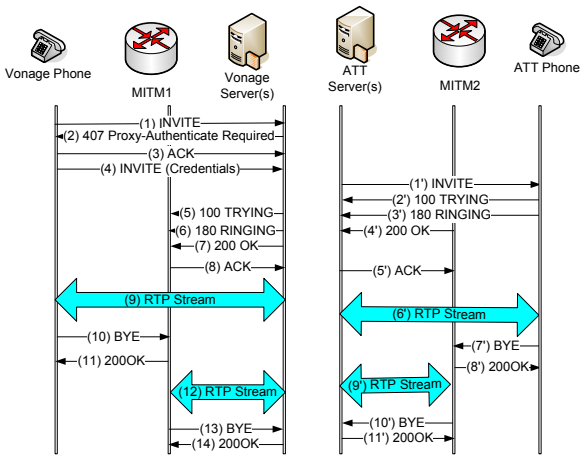

Figure 7: Message Flow of ByeDelay Billing Attack

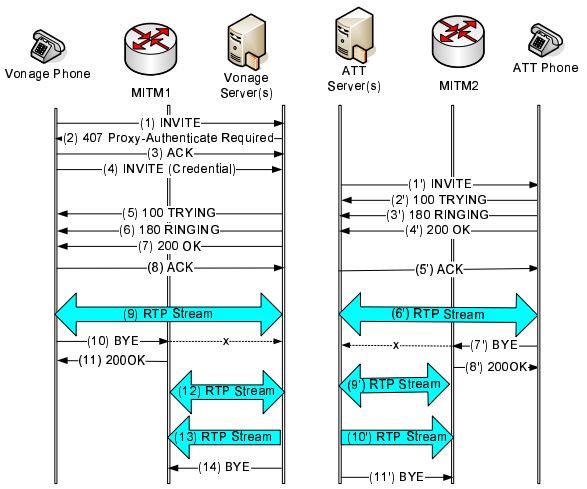

Figure 8: Message Flow of ByeDrop Billing Attack

3.4 ByeDrop Billing Attack

ByeDrop billing attack prolongs the duration of established calls between targeted VoIP subscribers by simply dropping the `BYE` messages. Figure 8 shows the message flow of ByeDrop billing attack. In our experiments, the normal call lasted for about 2 minutes before one side hung up. Similar to ByeDelay billing attack, the MITMs intercepted the `BYE` messages, and they replied with `200 OK` messages which gave the caller and callee the impression that the call had terminated successfully. The bogus RTP streams, as shown in step 12 and 9', lasted for about 20 minutes before the MITM2 stopped sending RTP streams. Surprisingly, both Vonage's RTP server and AT&T's RTP server kept sending unidirectional RTP streams to Vonage phone and AT&T phone respectively for about 218 minutes. After replaying those RTP streams, we found out that they are just background sounds. After about 218 minutes, the Vonage SIP server and AT&T SIP server sent `BYE` messages (shown in step 14 and 10') to terminate the call and their corresponding RTP servers stopped sending RTP streams. We checked the Vonage's online call activity system, and it showed that the call between the Vonage phone and the AT&T phone lasted for about 240 minutes even though the real call between the Vonage phone and the AT&T phone was only 2 minutes long.

3.5 Discussion

We have shown that attackers, who are in between the SIP phone and SIP servers, could successfully launch billing attacks on subscribers of SIP-based VoIP services without the knowledge of the secret password shared between the SIP phones and the SIP servers. Since the SIP authentication mechanism has built-in anti-replay capability, any correct implementation of SIP authentication (e.g, Vonage) should immune from the InviteReplay billing attack. Therefore, AT&T's vulnerability to InviteReplay billing attack is due to its implementation error, and it is easy to fix. However, FakeBusy, ByeDelay and ByeDrop billing attacks exploit the inherent vulnerabilities of the existing SIP protocol, and they are more difficult to defend against. For example, correlation of the RTP streams and the SIP messages has been shown to be able to detect some VoIP attacks [24, 23, 22]. However, simply correlating the RTP streams and the SIP messages will not detect the FakeBusy, ByeDelay and ByeDrop billing attacks which could generate bogus RTP streams with correct IP addresses, port numbers and sequence numbers. This is due to the lack of integrity protection of the SIP message and RTP stream. If the SIP messages are fully protected and all the RTP streams are properly encrypted with anti-replay protection, the FakeBusy and ByeDelay billing attacks could be detected and prevented. However, ByeDrop billing attack is still viable even if all the SIP messages are fully protected and all the RTP streams are properly encrypted. Since packet dropping could indeed happen naturally on the Internet, it is quite difficult to differentiate the ByeDrop billing attack from the natural packet loss. How to effectively mitigate such kind of billing attacks remains an open research problem.

4 Related Works

Here we briefly overview existing works related to SIP security. Arkko et al [3] proposed a scheme to negotiate the security mechanism used between a UA and and its next-hop SIP entity. Baugher et al [4] proposed Secure Real-time Transport Protocol (SRTP) to protect the RTP traffic. Salsano et al [20] evaluated the SIP process-

ing overhead of SIP authentication and TLS. Geneiatakis et al [8] looked at the potential threats to SIP. McGann and Sicker [12] analyzed detection capability of several VoIP security tools. Reynolds and Goshal [16] proposed multi-protocol protection against flooding attacks against VoIP network. Rosenberg [18] described the possibility to trick the *interactive voice response* (IVR) systems into sending large amount of RTP packets to the target, and proposed using *interactive connectivity establishment* (ICE) to mitigate such a denial-of-service attack. Wu et al [24] and Sengar et al. [23, 22] proposed cross-protocol methods to detect denial-of-service attacks on VoIP. However, none of the previously proposed VoIP intrusion detection methods could detect the billing attacks we have described.

The concept of billing attack on VoIP is not new. Both Internet draft [13] and paper [24] have mentioned the possibility of billing attacks on SIP-based VoIP. However, they have neither implemented nor validated the possible billing attacks. To the best of our knowledge, ours is the first published work that actually implements and empirically validates billing attacks against deployed SIP-based VoIP.

5 Conclusions

Billing is fundamental to any commercial VoIP services and it has direct impact on each commercial VoIP subscriber. The use of public Internet for signaling makes VoIP more susceptible to signaling-based billing attacks than traditional PSTN calls. In addition, the change from PSTN architecture, where the complexity and intelligence are in the network core, to VoIP, where the the complexity and intelligence are moved to edge handsets with relatively dumb network core, may inherently make the VoIP billing and accounting more challenging for the VoIP service providers. We have presented four billing attacks on subscribers of SIP-based VoIP services: InviteReplay attack, FakeBusy attack, ByeDelay attack and ByeDrop attack, which could incur charges on calls not made by the subscribers or overcharges on calls made by the subscribers. Our experiments show that millions subscribers of leading commercial VoIP service providers such as Vonage and AT&T are vulnerable to various billing attacks, and the billing of existing SIP-based VoIP services is not trustworthy. We suspect that the VoIP billing vulnerabilities we have identified are just the tip of the iceberg. We hope that our work will bring the the research community's attention to the problems of VoIP billing and will inspire future work on securing VoIP billing.

6 Acknowledgments

The authors would like to thank the anonymous reviewers for their insightful comments that helped to improve the presentation of this paper. This work was partially supported by NSF grant CNS-0524286.

References

[1] Vonage subscriber lines. URL. http://www.vonage.com/corporate/index.php?lid=footer_corporate.

[2] F. Andreasen and B. Foster. Media Gateway Control Protocol (MGCP) Version 1.0. *RFC 3435, IETF*, January 2003.

[3] 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.

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

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

[6] T. Dierks and C. Allen. The TLS Protocol. *RFC 2246, IETF*, January 1999

[7] J. Franks, P. Hallam-Baker, J. Hostetler, S. Lawrence, P. Leach, A. Luotonen and L. Stewart. HTTP Authentication: Basic and Digest Access Authentication. *RFC 2617, IETF*, June 1999.

[8] D. Geneiatakis, G. Kambourakis, T. Dagiuklas, C. Lambrinoudakis and S. Gritzalis. SIP Security Mechanisms: A State-of-the-art Review. In *the Proceedings of the Fifth International Network Conference (INC 2005)*, pages 147–155, July 2005, Samos, Greece,

[9] AT&T's CallVantage. URL. https://www.callvantage.att.com/.

[10] M. Handley and V. Jacobson. SDP: Session Description Protocol. *RFC 2327, IETF*, April 1998.

[11] S. Kent and R. Atkinson. Security Architecture for the Internet Protocol. *RFC 2401, IETF*, November 1998.

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

[13] S. Niccolini, E. Chen. VoIP Security Threats relevant to SPEERMINT. http://tools.ietf.org/html/draft-niccolini-speermint-voipthreats-01. March 2007.

[14] B. Ramsdell, Editor. S/MIME Version 3 Message Specification. *RFC 2633, IETF*, June 1999.

[15] ITU-T Recommendation H.323v.4 Packet-based multimedia communications systems. November 2000.

[16] 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.

[17] R. Rivest. The MD5 Message-Digest Algorithm. *RFC 1321, IETF*, April 1992.

[18] J. Rosenberg. The Real Time Transport Protocol (RTP) Denial of Service (Dos) Attack and its Prevention. http://www-rn.informatik.uni-bremen.de/ietf/mmusic/id/draft-rosenberg-mmusic-rtp-denialofservice-00.txt. June 2003.

[19] 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.

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

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

[22] H. Sengar, H. Wang, D. Wijesekera, and S. Jajodia. Fast Detection of Denial of Service Attacks on IP Telephony. In *Proceedings of the 14th IEEE International Workshop on Quality of Service (IWQoS 2006)*, June 2006.

[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] 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.