Extracting Evidence Related to VoIP Calls

David Irwin, Jill Slay

Chapter 17

EXTRACTING EVIDENCE RELATED TO VoIP CALLS

David Irwin and Jill Slay

Abstract The Voice over Internet Protocol (VoIP) is designed for voice communications over IP networks. To use a VoIP service, an individual only needs a user name for identification. In comparison, the public switched telephone network requires detailed information from a user before creating an account. The limited identity information requirement makes VoIP calls appealing to criminals. In addition, due to VoIP call encryption, conventional eavesdropping and wiretapping methods are ineffective. Forensic investigators thus require alternative methods for recovering evidence related to VoIP calls. This paper describes a digital forensic tool that extracts and analyzes VoIP packets from computers used to make VoIP calls.

Keywords: VoIP calls, packet extraction, packet analysis

1. Introduction

Voice over Internet Protocol (VoIP) telephony is an inexpensive and increasingly popular alternative to using traditional telephone networks. The use of VoIP in U.S. businesses is expected to reach 79% by 2013 [2]. Meanwhile, the lack of technology for law enforcement to monitor VoIP calls, the low barrier for entry and the anonymity provided by VoIP service are making it very attractive to criminals [3].

Fortunately, the remnants of a VoIP call remain in the physical memory of the computers used for the call. The information available includes signaling information, the digitized call, and information about the VoIP client. The signaling information is related to the setup and initialization of the VoIP call. The digitized call comprises packets that contain the encapsulated voice component. Information specific to the VoIP application being used, such as the contact list, is also saved. It

is possible to manually search for known VoIP remnants, but this is a time-consuming process that requires considerable expertise.

McKemmish [4] defines digital forensics as "the process of identifying, preserving, analyzing and presenting digital evidence in a manner that is legally acceptable." The digital forensic search tool described in this paper is designed to support all four steps. A byte-for-byte copy of the original memory is created without modifying the original digital evidence. This evidence is processed and formatted into a human-readable format for presentation in a court of law.

Several researchers have investigated memory forensic techniques for extracting evidence related to VoIP calls [6, 7, 9]. This paper builds on this work by describing a forensic tool that detects and reconstructs VoIP packet sequences from a computer memory capture. In addition, it provides a means for extracting user information and VoIP client information. Experimental tests demonstrate that the tool locates more than 97% of the packets in VoIP calls.

2. Internet Protocol

VoIP is a collection of several protocols that set up, maintain and tear down calls involving the encapsulation and transportation of voice packets over the Internet. The two most prominent protocols used are the User Datagram Protocol (UDP) and the Real-Time Transport Protocol (RTP). UDP is a transport layer protocol used by Skype [8]. RTP is an application layer protocol that, in the case of X-Lite [1], uses UDP as the transport layer protocol. Both these protocols include an Ethernet frame link layer and an Internet Protocol (IP) Internet layer header.

IP commonly uses version 2 Ethernet frames. An Ethernet frame consists of a seven-byte preamble, a one-byte start of frame delimiter, and two six-byte Media Access Control (MAC) headers, one each for the source and destination. Following the MAC headers are the two-byte Ethertype, the IP/UDP/RTP data payload in bytes 46 to 1,500, and a four-byte cyclic redundancy checksum for packet integrity.

IP provides Internet addresses in its headers, allowing packets to be routed from their source to a destination IP address. However, an IP address is not sufficient to deliver an IP packet from a source IP address to a destination IP address. The port numbers of the source and destination computers must also be known for a VoIP call to take place. UDP maintains the port information. While UDP does not guarantee IP packet delivery, it is well-suited to VoIP because of the real-time nature of voice communications. Thus, VoIP uses the IP/UDP protocol stack shown in Figure 1.

IP Bit Offset	0-3	4-7	8-13	14-15	16-18	19-31
0	Version	Header Length	DSCP	ECN	Total Length	
32	Identification				Flags	Fragment Offset
64	Time to Live		Protocol		Header Checksum	
96	Source Address					
128	Destination Address					
UDP Bit Offset	0-15				16-31	
160	Source Port Number				Destination Port Number	
192	Length				Checksum	
224	Payload					

Figure 1. IP and UDP packet headers.

RTP [5] provides transport for real-time applications that transmit audio over packet-switched networks. The protocol incorporates information such as packet sequence numbers and timestamps. This allows a receiving application to buffer and sequence packets in the correct order for audio playback. Thus, the complete VoIP stack is IP/UDP/RTP.

Bit Offset	0-1	2	3	4-7	8	9-15	16-31
0	Version	Padding	Ext.	CSRC Count	Marker	Payload Type	Sequence Number
32	Timestamp						
64	Synchronization Source (SSRC) Identifier						
96	Contributing Source (CSRC) Identifier						
96+32*CC	Payload						

Figure 2. RTP packet header.

Figure 2 presents the RTP packet header format. In RTP, the Synchronization Source (SSRC) field identifies the source of the synchronization (e.g., computer clock). The Contributing Source (CSRC) field identifies the source of the individual contributions that make up the single data stream payload for the packet. It is not necessary to use RTP to participate in a VoIP call. VoIP applications such as Skype do not use RTP; X-Lite, on the other hand, uses RTP. RTP provides a means for a VoIP client to reassemble and synchronize packets.

```
-Request-Line:
-Invite sip:8889215862@sip.pennytel.com
        Method: Invite
-Message Header
        Contact:sip:8889215864@119.40.108.72:26610
-To: "david"< sip:8889215862@sip.pennytel.com>
        SIP Display info: "david"
-SIP to address:sip:8889215862@sip.pennytel.com
        SIP to address User Part: 8889215862
        SIP to address Host Part: sip.pennytel.com
-From:"8889215864"sip:8889215864@sip.pennytel.com
```

Figure 3. SIP invite request.

3. VoIP Packet Identification

After an individual registers with a VoIP service provider, the individual uses the provider's client to make calls. To initiate a call, the client connects to the provider using the Session Initiation Protocol (SIP). Figure 3 shows a SIP invite request. Elements of a SIP invite request that are important to a forensic investigator include the user's registered name (`david`), unique SIP user identifier (`8889215862`) and host (`sip.pennytel.com`).

SIP contains information about the call participants based on their unique SIP identifiers. A regular expression search can be used to identify VoIP packets in a memory capture. In our experiments, we used a hex editor to search two physical memory captures. The first capture was made after a Skype call that only uses UDP. The second was made after an X-Lite VoIP call that uses RTP and SIP.

```
0000  00 0C 29 B6 57 76 00 21  6A 4A D6 26 08 00 45 00
0010  00 7D 58 77 00 00 80 11  5F DD C0 A8 00 66 C0 A8
0020  00 65 A1 01 53 CD 00 69  89 E6 80 6B 23 01 00 2B
0030  59 1C 22 14 AD 31 3C 64  7B 82 29 6C E0 18 DD A9
0040  25 EA 44 65 61 9A C1 66  D3 A1 B9 09 BC 38 B1 86
0050  89 66 63 11 D2 44 5F 88  A3 2D E4 63 8E A5 B8 73
0060  26 41 09 BD 90 99 65 1D  E7 1B 85 D6 A3 A6 5A 09
0070  DC 21 5C C0 A8 39 05 BB  F1 A5 1B E6 A2 29 4A E0
0080  6C 56 92 47 9D CA 65 00
```

Figure 4. VoIP frame with the search expression highlighted.

Figure 4 shows a single VoIP packet capture with the search pattern highlighted. The headers are segmented by vertical lines (Ethernet/IP/UDP/RDP/RTP/Payload).

The Ethernet frame is fourteen bytes in length (the preamble and start of the frame delimiter are not shown) with six bytes each for the

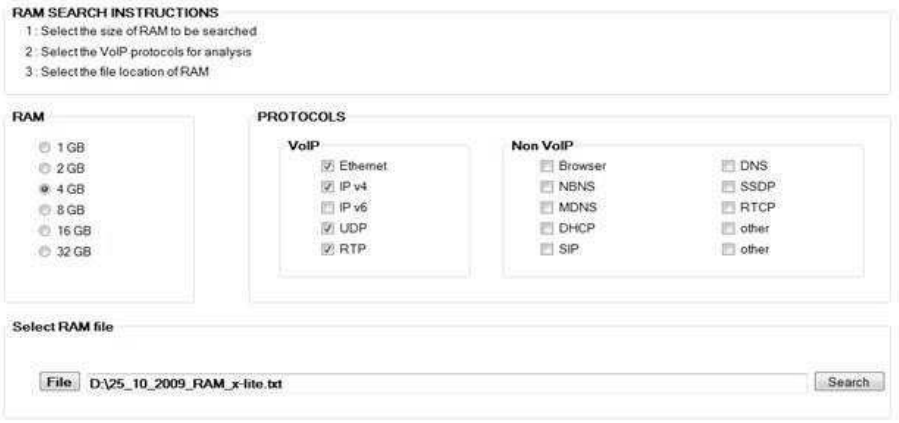

Figure 5. User interface.

source and destination MAC addresses. The last two bytes identifies the Ethertype, which, in the case of VoIP, is `0x0800` for an IP packet.

During our analysis of a 4 GB memory capture, a search using the IP identifier (`0x0800`) and the first byte of the IP header (`0x45`) corresponding to the byte search pattern `0x080045` yielded 8,881 hits.

The UDP header is eight bytes long and does not form part of the search pattern. The identification of UDP verifies the use of an Ethernet/IP/UDP stack and the port numbers identify the two parties involved in a call. The UDP protocol is identified by byte 10 of the IP header (`0x11`), indicating that the next protocol is in fact UDP. The search pattern `0x080045--------11` yielded 559 hits. If the VoIP client uses RTP, then the RTP header follows the UDP header.

The identification of RTP is accomplished by examining the first byte that follows the UDP header. The first byte of RTP contains the version number, padding bit, extension bit and CSRC count. In the example in Figure 4, the current SIP version is 2 and the other elements are predominantly empty; thus, the byte has the value `0x80`.

A search of the 4 GB memory capture using the complete pattern `0x080045--------11----------------80` yielded no false positive errors.

4. Forensic Tool

The forensic tool implemented to extract and analyze VoIP packets has a simplified interface with tabbed browsing and asynchronous functionality (Figure 5). The user first selects the memory capture file to be searched. By default, the Ethernet Protocol, IP version 4, UDP and

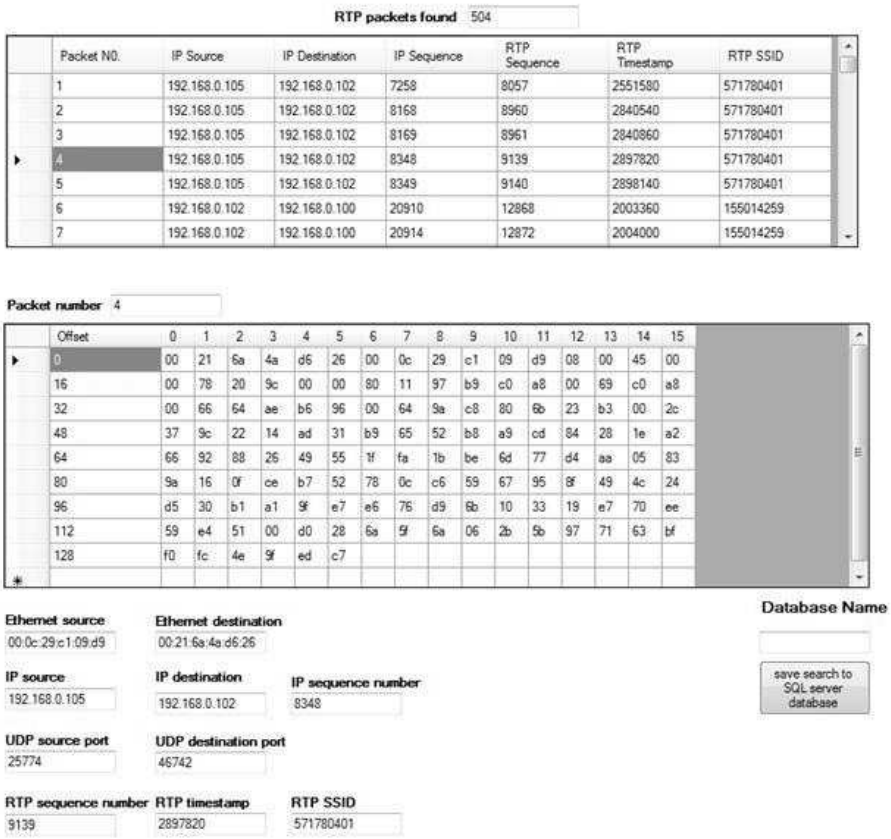

Figure 6. Results interface.

RTP are automatically selected and searched. The option to search for IP version 6 is also available. The search looks for VoIP packets that match the pattern with and without the RTP component.

Figure 6 presents the results of an analysis of a memory capture using the forensic tool. The top data grid displays the RTP packets recovered. When a user selects an individual packet, the bottom grid updates to display detailed packet information. The displayed information includes the Ethernet source and destination addresses, IP source and destination addresses and sequence number, UDP source and destination port numbers and sequence number, timestamp, and synchronization source identifier. The recovered packets can be saved for further analysis in an SQL Server database. For example, the payloads can recombined into a single stream and attempts can be made to decrypt the payloads either by brute force or with the assistance of the VoIP provider.

Table 1. Wireshark and RAM recovery results.

VoIP Client	Duration (seconds)	Wireshark Packet Count	RAM Packets (Total)	RAM Packets (Unique)	% Call Recovered
Skype	180	18,701	41,959	18,208	97.4%
X-Lite	30	3,097	4,759,	3,093	99.9%
X-Lite	30	3,290	5,488	3,274	99.5%
X-Lite	180	9,089	17,695	9,063	99.7%

The forensic tool can also be used to train forensic analysts. An individual packet may be expanded and each protocol highlighted in a different color to facilitate the interpretation of individual protocol fields. The color-coded graphical representation of the VoIP protocol stack greatly simplifies the interpretation and understanding of the overall frame and the individual protocols.

5. Experimental Results

Two memory captures were performed after VoIP calls. The first was a 4 GB RAM capture performed after a Skype call that lasted three minutes. The second memory capture occurred after a clean restart on the following day and three successive X-Lite calls, the first lasting 30 seconds, the second 30 seconds and the third three minutes.

Table 1 compares the remnants of the calls recovered by the digital forensic tool (RAM capture) versus the Wireshark capture of the VoIP calls. Note that the total number of packets recovered by the RAM capture (Total) exceeds the actual number of packets in the call (Wireshark Packet Count). It was found that duplicate packets exist in up to six different locations in memory. Filtering these duplicate packets provides a more accurate measure of the number of packets recovered (Unique). The forensic tool locates nearly all the VoIP packets corresponding to the two types of calls. Note that the recovery percentage is lower for the Skype call because it does not use RTP and, therefore, does not benefit from the use of a longer search expression.

6. Conclusions

The forensic tool presented in this paper successfully recovers VoIP packets from memory captures. The tool also helps extract user details from VoIP application control signals. The ability to analyze, store and format VoIP packets is particularly valuable in forensic investigations.

Several opportunities exist to improve the tool. For example, before transmission and during playback, the call is in an unencrypted form. Therefore, the potential exists to extract unencrypted audio from memory. Another enhancement involves the creation of a database with contact list structures and control signal information associated with commonly used VoIP clients.

Acknowledgements

This research was supported by the Australian Research Council via Linkage Grant LP0989890 and by the Australian Federal Police.

References

[1] CounterPath Corporation, X-Lite, Vancouver, Canada (www.counterpath.com/x-lite.html).

[2] In-Stat, VoIP penetration forecast to reach 79% of U.S. businesses by 2013, Scottsdale, Arizona (www.instat.com/newmk.asp?ID=2721), February 2, 2010.

[3] R. Koch, Criminal activity through VoIP: Addressing the misuse of your network, Technology Marketing Corporation, Norwalk, Connecticut (www.tmcnet.com/voip/1205/special-focus-criminal-activity-through-voip.htm), 2010.

[4] R. McKemmish, What is forensic computing? *Trends and Issues in Crime and Criminal Justice*, no. 118, pp. 1–6, 1999.

[5] H. Schulzrinne, S. Casner, R. Frederick and V. Jacobson, RTP: A Transport Protocol for Real-Time Applications, RFC 3550, Internet Engineering Task Force, Fremont, California (tools.ietf.org/html/rfc3550), 2003.

[6] M. Simon and J. Slay, Voice over IP: Forensic computing implications, *Proceedings of the Fourth Australian Digital Forensics Conference*, pp. 1–6, 2006.

[7] M. Simon and J. Slay, Enhancement of forensic computing investigations through memory forensic techniques, *Proceedings of the International Conference on Availability, Reliability and Security*, pp. 995–1000, 2009.

[8] Skype, Luxembourg (www.skype.com).

[9] J. Slay and M. Simon, Voice over IP forensics, *Proceedings of the First International Conference on Forensic Applications and Techniques in Telecommunications, Information and Multimedia*, pp. 10:1–10:6, 2008.