

Volume

5

## MANAGING CALL FLOWS USING SIP

---

Mark A. Miller, P.E.  
President  
DigiNet Corporation

A technical briefing from:



June 2002

## Table of Contents

<b>Executive Summary</b>	<b>i</b>
<b>1. SIP Architecture</b>	<b>1</b>
<b>2. SIP Messages</b>	<b>2</b>
<b>3. SIP-related Protocols</b>	<b>3</b>
<b>4. Case Study: Analyzing SIP Phone Connections</b>	<b>4</b>
<b>5. Looking Ahead</b>	<b>16</b>
<b>6. Acronyms and Abbreviations</b>	<b>17</b>
<b>7. About the Author and Sponsor</b>	<b>18</b>

## Executive Summary

This is the fifth of six technical briefing papers that examine the concepts, operation and analysis of *converged networks*. Specific protocols have been developed to manage the call establishment and disconnect procedures within these converged networks. Those procedures are known as *signaling*, and the protocol described in this paper, known as the Session Initiation Protocol, or SIP, is a signaling protocol defined by the Internet Engineering Task Force (IETF). SIP has been implemented within a number of systems, including Internet Protocol (IP)-based telephones, IP-based Private Branch Exchanges (PBXs), and carrier networks. SIP provides many of the functions defined in the H.323 standard developed by the International Telecommunication Union — Telecommunication Standards Sector (ITU-T), but with lower complexity and overhead.

A SIP-based network includes two key elements: user agents and servers. The user agents exist on an end system, such as a SIP-based telephone, and can establish connections with other SIP peer devices. Four different types of servers, which may be logical (instead of physical) devices, are also defined within the SIP architecture. The proxy server acts as an intermediary, forwarding requests on behalf of other devices. The redirect server directs a client to the appropriate location to complete a task. The registrar server keeps track of end stations in conjunction with a database known as a location server (or the location service).

SIP relies upon other IETF and ITU-T protocols for specific functions. The Session Description Protocol (SDP) provides a standard way of describing multimedia sessions. The Session Announcement Protocol (SAP) periodically announces conference session parameters. The Real-Time Transport Protocol (RTP) and Real-Time Control Protocol (RTCP) provide information on transport and session management. Finally, ITU-T standard voice and video encoding algorithms, such as G.723.1, provide the analog-to-digital conversions and signal compression.

A case study, showing telephone connections between two SIP devices, will illustrate the operation and analysis of these protocols in greater detail.

# 1. SIP Architecture

SIP is a peer-to-peer signaling protocol, developed by the Internet Engineering Task Force (IETF), that allows end devices to initiate and terminate communication sessions. This protocol is defined in RFC 2543 and incorporates elements from other IETF-developed protocols, including the Hypertext Transfer Protocol (HTTP) described in RFC 2068, the Simple Mail Transfer Protocol (SMTP) described in RFC 2821, and the Session Description Protocol (SDP) described in RFC 2327.

The SIP architecture includes two key components: *user agents* and *servers*. The user agent represents an end system and contains two subsystems: a *user agent client* (UAC) which generates requests, and a *user agent server* (UAS) which responds to requests. These two elements are shown in Figure 1.

SIP servers are entities that are logical functions, not necessarily separate physical devices. Four of these server functions are defined:

- Proxy servers: are network hosts that act as intermediaries for the purpose of making requests on behalf of other clients. Proxies must therefore act in both client and server roles — they route SIP requests to user agent servers, and route SIP responses to user agent clients. The proxy server also performs a routing function, ensuring that requests are forwarded to the appropriate entity, and may also enforce policies, such as assuring that a particular user is allowed to place the requested call.
- Redirect servers: are logical entities that direct a client to an alternate set of uniform resource indicators (URIs) for completion of the requested task.
- Registrar servers: receive and process registration messages that allow the location of end stations to be tracked. This registration service works in conjunction with the location service noted below.
- Location servers: provide an abstract database service that binds an address with a particular network domain. This server works in conjunction with the registration service and allows a proxy server to input a URI and receive a set of URIs that tells it where to send a particular request.

The various SIP server functions are also illustrated in Figure 1.



- INVITE: indicates that the user or service is being invited to participate in a session. The body of this message would include a description of the session to which the callee is being invited.
- ACK: confirms that the client has received a final response to an INVITE request, and is only used with INVITE requests.
- OPTIONS: is used to query a server about its capabilities.
- BYE: is sent by a user agent client to indicate to the server that it wishes to terminate the session.
- CANCEL: is used to cancel a pending request.
- REGISTER: is used by a client to register contact information.

The response messages contain Status Codes and Reason Phrases that indicate the current condition of this request. The status code values, which are similar to those used with HTTP, are divided into six general categories:

- 1xx: Provisional — the request has been received and processing is continuing.
- 2xx: Success — the action was successfully received, understood and accepted.
- 3xx: Redirection — further action is required to process this request.
- 4xx: Client Error — the request contains bad syntax, and cannot be fulfilled at this server.
- 5xx: Server Error — the server failed to fulfill an apparently valid request.
- 6xx: Global Failure — the request cannot be fulfilled at any server.

Specific details on the SIP message formats, status codes, and other parameters are found in RFC 2543.

## **3. SIP-related Protocols**

SIP incorporates elements from several other IETF and ITU-T standards:

- IETF Session Description Protocol (SDP) which defines a standard method of describing the characteristics of a multimedia session.
- IETF Session Announcement Protocol (SAP), which periodically announces parameters of a conference session.
- IETF Real-Time Transport Protocol (RTP) and Real-Time Control Protocol (RTCP) provide information on transport and session management. RTP is the protocol within the H.323 suite that carries the digitally encoded voice or video packets between the end stations. RTCP manages the sessions by periodically transmitting packets containing feedback on the quality of the data distribution.
- ITU-T recommended encoding algorithms, such as G.723.1, G.728, and G.729 for audio encoding, or H.261 or H.263 for video encoding.

Note that the Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) are used in support of these protocols, as shown in Figure 2.

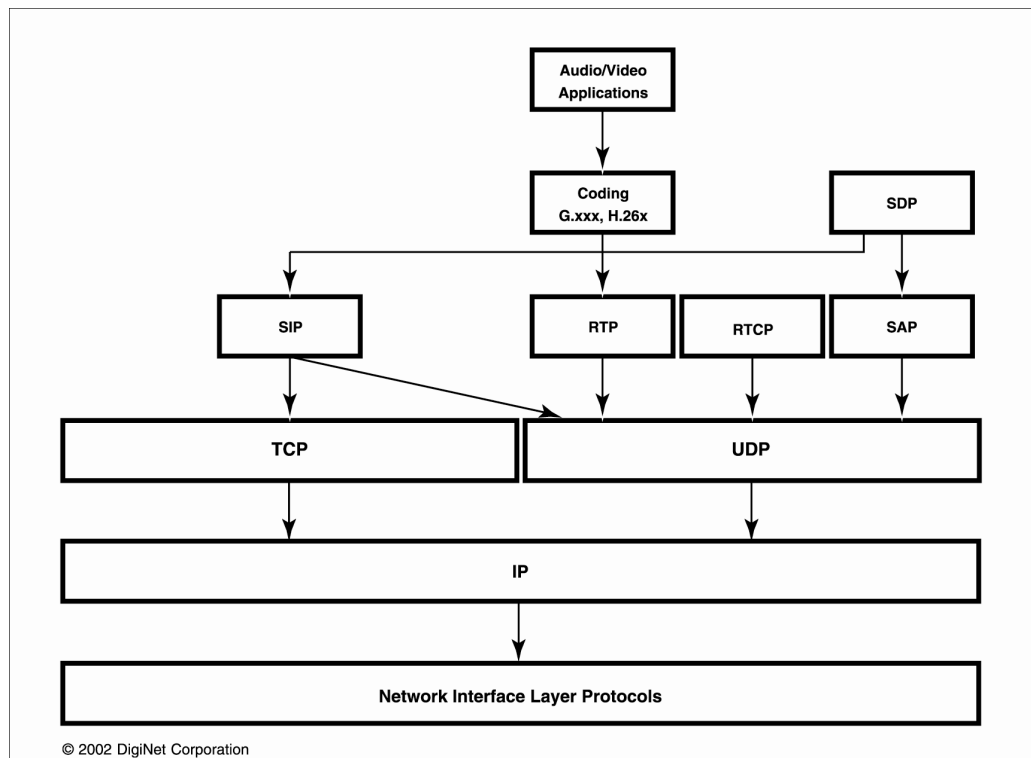


FIGURE 2. **SIP-related Protocols**

## 4. Case Study: Analyzing SIP Phone Connections

As the previous sections have detailed, SIP provides the signaling functions necessary to establish, manage, and terminate a connection between multimedia endpoints. As an example of SIP protocol analysis, consider the network shown in Figure 3, in which an analog telephone connected to a Cisco Systems, Inc. SIP gateway initiates a call to a Cisco SIP telephone connected to the same network.

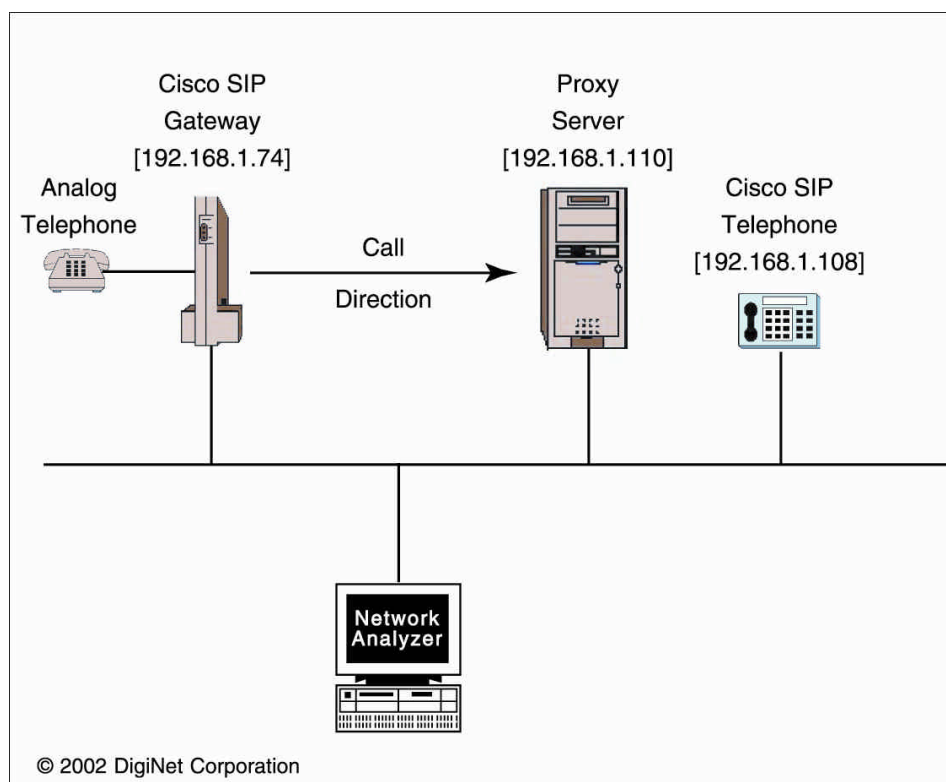


FIGURE 3. **SIP Phone-to-Phone Connection**

Note that a proxy server exists on this network and will be involved in the call establishment and termination procedures. The SIP control messages will flow between the SIP devices and the proxy server, and the RTP media information (voice samples) will flow between the SIP gateway and the SIP telephone (Figure 4A).

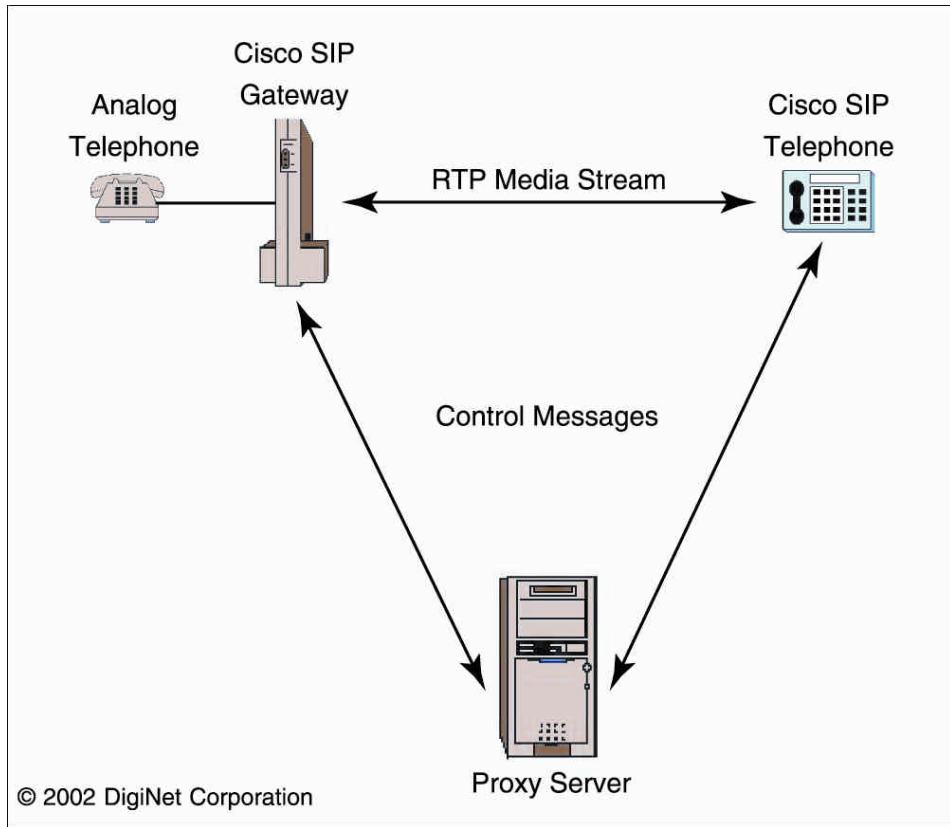


FIGURE 4A. **SIP Control Message Flow**

The SIP control messages can be broken down into three general categories. These functions are detailed in Figure 4B and include the direction of the information flow:

1. Connection establishment using SIP (Frames 1–6, 129–131)
2. Information transfer using RTP (Frames 131–225)
3. Connection termination using SIP (Frames 226–229)

For brevity, some of the RTP voice traffic (Frames 131–225) has been filtered out, and is not shown in either Figure 4B or the accompanying Trace 1.

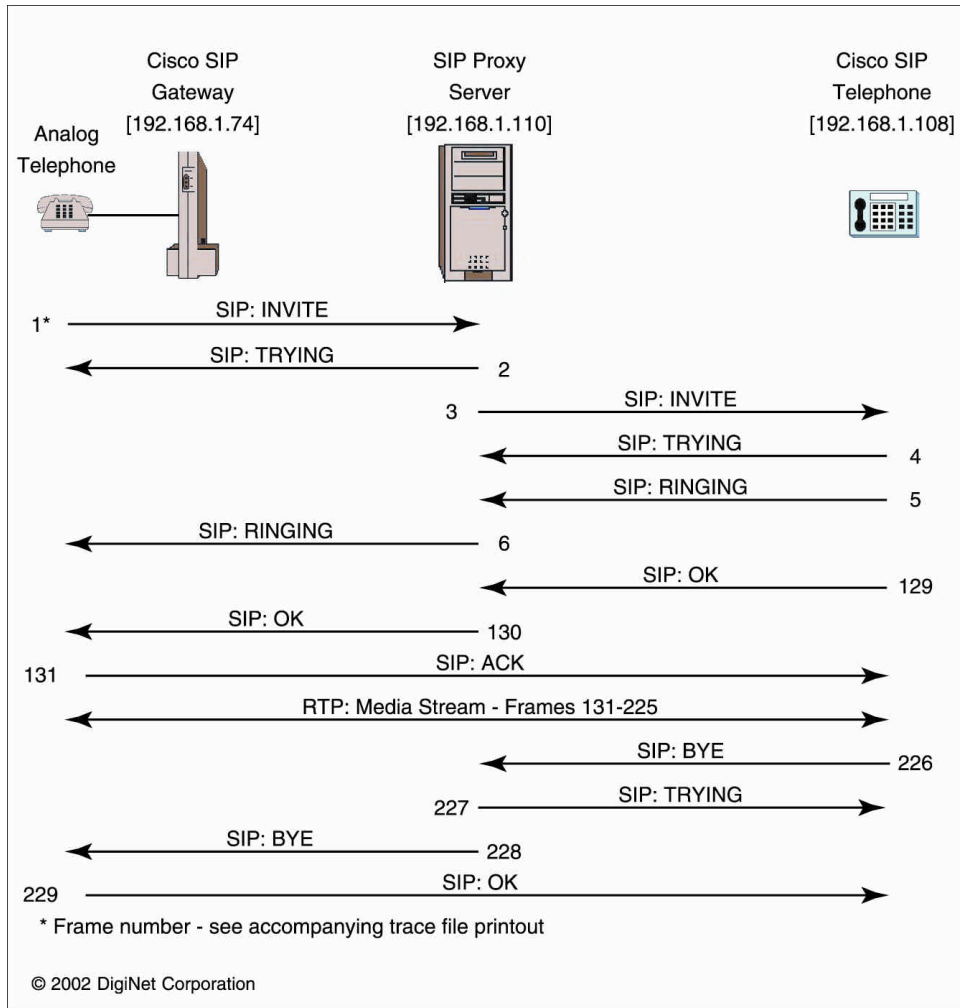


FIGURE 4B. SIP Control Message Details

The details of these three functions are shown in Trace 1, provided by output from the Network Associates, Inc. *Sniffer*® analyzer. Trace 1 provides four columns of information: the Frame Number, the Source Station, the Destination Station, and a brief Summary of the protocol operations within that frame. Frames of special interest, which will be discussed further, are shown in boldface type. For brevity, traffic not relevant to our discussion has been filtered out, yielding some gaps in frame numbers.

Note that the inclusion of the proxy server in the network makes for a two-step communication process. For example, the SIP INVITE message is generated in Frame 1, and then passed to the desired destination in Frame 3. Similarly, the SIP RINGING response is sent to the proxy server in Frame 5, and then passed to the gateway in Frame 6. After the connection establishment is confirmed in Frame 130, then direct communication between the two endpoints begins.

TRACE 1. SIP Phone-to-Phone Connection Summary

Frame	Source	Destination	Summary
1	Cisco SIP Gateway	SIP Proxy Server	<b>SIP: C INVITE</b> sip:108@192.168.1.110; user=phone;phone context=national SIP/2.0
2	SIP Proxy Server	Cisco SIP Gateway	<b>SIP: R SIP/2.0 Response</b> Status Code=100 (Informational) Response-Phrase=Trying
3	SIP Proxy Server	Cisco SIP Phone 108	<b>SIP: C INVITE</b> sip:108@192.168.1.108 SIP/2.0
4	Cisco SIP Phone 108	SIP Proxy Server	<b>SIP: R SIP/2.0</b> Response Status Code=100 (Informational) Response-Phrase=Trying
5	Cisco SIP Phone 108	SIP Proxy Server	<b>SIP: R SIP/2.0 Response</b> Status Code=180 (Informational) Response-Phrase=Ringing
6	SIP Proxy Server	Cisco SIP Gateway	<b>SIP: R SIP/2.0 Response</b> Status Code=180 (Informational) Response-Phrase=Ringing
.	.	.	.
129	Cisco SIP Phone 108	SIP Proxy Server	<b>SIP: R SIP/2.0 Response</b> Status Code=200 (Success) Response-Phrase=OK
130	SIP Proxy Server	Cisco SIP Gateway	<b>SIP: R SIP/2.0 Response</b> Status Code=200 (Success) Response-Phrase=OK
131	Cisco SIP Gateway	Cisco SIP Phone 108	<b>SIP: C ACK</b> sip:108@192.168.1.108:5060 SIP/2.0
132	Cisco SIP Gateway	Cisco SIP Phone 108	RTP: Payload=PCMU audio SEQ=699 SSRC=552665418
133	Cisco SIP Gateway	Cisco SIP Phone 108	RTP: Payload=PCMU audio SEQ=700 SSRC=552665418
134	Cisco SIP Phone 108	Cisco SIP Gateway	RTP: Payload=PCMU audio SEQ=4067 SSRC=490861163
135	Cisco SIP Gateway	Cisco SIP Phone 108	RTP: Payload=PCMU audio SEQ=701 SSRC=552665418
136	Cisco SIP Phone 108	Cisco SIP Gateway	RTP: Payload=PCMU audio SEQ=4068 SSRC=490861163
137	Cisco SIP Gateway	Cisco SIP Phone 108	RTP: Payload=PCMU audio SEQ=702 SSRC=552665418
138	Cisco SIP Phone 108	Cisco SIP Gateway	<b>RTP: Payload=PCMU audio</b> SEQ=4069 SSRC=490861163
139	Cisco SIP Gateway	Cisco SIP Phone 108	RTP: Payload=PCMU audio SEQ=703 SSRC=552665418
140	Cisco SIP Phone 108	Cisco SIP Gateway	RTP: Payload=PCMU audio SEQ=4070 SSRC=490861163
.	.	.	.
220	Cisco SIP Phone 108	Cisco SIP Gateway	RTP: Payload=PCMU audio SEQ=4109 SSRC=490861163
221	Cisco SIP Gateway	Cisco SIP Phone 108	RTP: Payload=PCMU audio SEQ=743 SSRC=552665418

222	Cisco SIP Phone 108	Cisco SIP Gateway	RTP: Payload=PCMU audio SEQ=4110 SSRC=490861163
223	Cisco SIP Gateway	Cisco SIP Phone 108	RTP: Payload=PCMU audio SEQ=744 SSRC=552665418
224	Cisco SIP Phone 108	Cisco SIP Gateway	RTP: Payload=PCMU audio SEQ=4111 SSRC=490861163
225	Cisco SIP Gateway	Cisco SIP Phone 108	RTP: Payload=PCMU audio SEQ=745 SSRC=552665418
226	Cisco SIP Phone 108	SIP Proxy Server	<b>SIP: C BYE</b> sip:14083465118 @192.168.1.74:5060; user=phone SIP/2.0
227	SIP Proxy Server	Cisco SIP Phone 108	<b>SIP: R SIP/2.0 Response</b> Status Code=100 (Informational) Response-Phrase=Trying
228	SIP Proxy Server	Cisco SIP Gateway	<b>SIP: C BYE sip:14083465118</b> @192.168.1.74; user=phone SIP/2.0
229	Cisco SIP Gateway	Cisco SIP Phone 108	<b>SIP: R SIP/2.0 Response</b> Status Code=200 (Success) Response-Phrase=OK

The details of the SIP call establishment frames are shown in Trace 2, also taken from the Sniffer analyzer. Note that this form of the trace file includes the details of every field within the frame of interest.

Frame 1 is the SIP INVITE message, which identifies the Cisco SIP telephone (extension 108) that is being called. Also included in the INVITE message is the Session Description Protocol (SDP) information, which provides details regarding the session owner/creator (Cisco Systems SIP GW), session name (SIP Call), connection information (Internet, IPv4 192.168.1.74), start and stop times (0 0, representing a permanent, unbounded call), media type (audio) and Audio/Video Profile (AVP 0, representing Pulse Code Modulation mu-Law (G.711) audio information). The INVITE message that is passed to the destination from the gateway includes a Via parameter, identifying that gateway as an intermediate message point.

Other SIP messages involved in the connection establishment sequence include the TRYING (Frames 2 and 4), RINGING (Frames 5 and 6), and successful operation acknowledgements (Frames 129, 130, and 131) that function similarly to processes on the more familiar analog telephone network.

## TRACE 2. SIP Call Establishment Message Details

----- Frame 1 -----

SIP: ----- Session Initiation Protocol -----

SIP:  
SIP: **INVITE sip =108@192.168.1.110; user=phone;phone-context=national SIP/2.0**  
SIP: Via = SIP/2.0/UDP 192.168.1.74:58440  
SIP: From = "14083465118" <sip:14083465118@192.168.1.74>  
SIP: To = <sip:108@192.168.1.110;user=phone;phone-context=national>  
SIP: Date = Wed, 03 Mar 1993 22:15:45 UTC  
SIP: Call-ID = 4E43092B-AD0ECA15-0-F13A2A8@192.168.1.74  
SIP: Cisco-Guid = 1313016107-2903427603-0-252945060  
SIP: User-Agent = Cisco VoIP Gateway/ IOS 12.x/ SIP enabled  
SIP: CSeq = 101 INVITE  
SIP: Max-Forwards = 6  
SIP: Timestamp = 731196945  
SIP: Contact = <sip:14083465118@192.168.1.74:5060;user=phone>  
SIP: Expires = 180  
SIP: Content-Type = application/sdp  
SIP: Content-Length = 134  
SIP:

SDP: ----- SDP Header -----

SDP:  
SDP: v(sdp protocol version) =0  
SDP: o(owner/creator and session identifier) =CiscoSystemsSIP-GW-UserAgent 7802 3827 IN IP4 192.168.1.74  
SDP: s(session name) =SIP Call  
SDP: c(connection information) =IN IP4 192.168.1.74  
SDP: t(<start time> <stop time>) =0 0  
SDP: m(media name and transport address) =audio 20080 RTP/AVP 0  
SDP:

----- Frame 2 -----

SIP: ----- Session Initiation Protocol -----

SIP:  
SIP: **R SIP/2.0 Response Status Code=100(Informational) Response-Phrase=Trying**  
SIP: Via = SIP/2.0/UDP 192.168.1.74:58440  
SIP: From = "14083465118" <sip:14083465118@192.168.1.74>  
SIP: To = <sip:108@192.168.1.110;user=phone;phone-context=national>  
SIP: Call-ID = 4E43092B-AD0ECA15-0-F13A2A8@192.168.1.74  
SIP: CSeq = 101 INVITE  
SIP: Content-Length = 0  
SIP: Server = IndigoSipServer/4.0  
SIP:

----- Frame 3 -----

SIP: ----- Session Initiation Protocol -----

SIP:  
SIP: INVITE sip =108@192.168.1.108 SIP/2.0  
SIP: **Via = SIP/2.0/UDP 192.168.1.110;branch=c45115398e3b9314d91685a56ec22dc-e531ecb88f58c3ad59dbb84c3c31e9**  
SIP: Via = SIP/2.0/UDP 192.168.1.74:58440  
SIP: From = "14083465118" <sip:14083465118@192.168.1.74>  
SIP: To = <sip:108@192.168.1.110;user=phone;phone-context=national>  
SIP: Date = Wed, 03 Mar 1993 22:15:45 UTC  
SIP: Call-ID = 4E43092B-AD0ECA15-0-F13A2A8@192.168.1.74  
SIP: User-Agent = Cisco VoIP Gateway/ IOS 12.x/ SIP enabled  
SIP: CSeq = 101 INVITE  
SIP: Max-Forwards = 5  
SIP: Timestamp = 731196945  
SIP: Contact = <sip:14083465118@192.168.1.74:5060;user=phone>  
SIP: Expires = 180  
SIP: Content-Type = application/sdp  
SIP: Content-Length = 134  
SIP: Record-Route = <sip:14083465118@192.168.1.74;user=phone;maddr=192.168.1.110;

ForkingID=c61532549d38a692ffcae34698f21c8>

SIP:  
SDP: ----- SDP Header -----  
SDP:  
SDP: v(sdp protocol version) =0  
SDP: o(owner/creator and session identifier) =CiscoSystemsSIP-GW-UserAgent 7802 3827 IN IP4 192.168.1.74  
SDP: s(session name) =SIP Call  
SDP: c(connection information) =IN IP4 192.168.1.74  
SDP: t(<start time> <stop time>) =0 0  
SDP: m(media name and transport address) =audio 20080 RTP/AVP 0  
SDP:

----- Frame 4 -----

SIP: ----- Session Initiation Protocol -----  
SIP:  
**SIP: R SIP/2.0 Response Status Code=100(Informational) Response-Phrase=Trying**  
SIP: Via = SIP/2.0/UDP 192.168.1.110;branch=c45115398e3b9314d91685a56ec22dc-e531ecb88f58c3ad59dbb84c3c31e9,  
SIP/2.0/UDP 192.168.1.74:58440  
SIP: From = "14083465118" <sip:14083465118@192.168.1.74>  
SIP: To = <sip:108@192.168.1.110;user=phone;phone-context=national>;tag=f26b030083a1d410-0  
SIP: Call-ID = 4E43092B-AD0ECA15-0-F13A2A8@192.168.1.74  
SIP: Server = Cisco IP Phone/ Rev. 1/ SIP enabled  
SIP: CSeq = 101 INVITE  
SIP: Content-Length = 0  
SIP:

----- Frame 5 -----

SIP: ----- Session Initiation Protocol -----  
SIP:  
**SIP: R SIP/2.0 Response Status Code=180(Informational) Response-Phrase=Ringin**  
SIP: Via = SIP/2.0/UDP 192.168.1.110;branch=c45115398e3b9314d91685a56ec22dc-e531ecb88f58c3ad59dbb84c3c31e9,  
SIP/2.0/UDP 192.168.1.74:58440  
SIP: From = "14083465118" <sip:14083465118@192.168.1.74>  
SIP: To = <sip:108@192.168.1.110;user=phone;phone-context=national>;tag=f26b030083a1d410-0  
SIP: Call-ID = 4E43092B-AD0ECA15-0-F13A2A8@192.168.1.74  
SIP: Server = Cisco IP Phone/ Rev. 1/ SIP enabled  
SIP: CSeq = 101 INVITE  
SIP: Content-Length = 0  
SIP:

----- Frame 6 -----

SIP: ----- Session Initiation Protocol -----  
SIP:  
**SIP: R SIP/2.0 Response Status Code=180(Informational) Response-Phrase=Ringin**  
SIP: Via = SIP/2.0/UDP 192.168.1.74:58440  
SIP: From = "14083465118" <sip:14083465118@192.168.1.74>  
SIP: To = <sip:108@192.168.1.110;user=phone;phone-context=national>;tag=f26b030083a1d410-0  
SIP: Call-ID = 4E43092B-AD0ECA15-0-F13A2A8@192.168.1.74  
SIP: Server = Cisco IP Phone/ Rev. 1/ SIP enabled  
SIP: CSeq = 101 INVITE  
SIP: Content-Length = 0  
SIP:

----- Frame 129 -----

SIP: ----- Session Initiation Protocol -----  
SIP:  
**SIP: R SIP/2.0 Response Status Code=200(Success) Response-Phrase=OK**  
SIP: Via = SIP/2.0/UDP 192.168.1.110;branch=c45115398e3b9314d91685a56ec22dc-e531ecb88f58c3ad59dbb84c3c31e9,  
SIP/2.0/UDP 192.168.1.74:58440  
SIP: From = "14083465118" <sip:14083465118@192.168.1.74>  
SIP: To = <sip:108@192.168.1.110;user=phone;phone-context=national>;tag=f26b030083a1d410-0

SIP: Call-ID = 4E43092B-AD0ECA15-0-F13A2A8@192.168.1.74  
SIP: Server = Cisco IP Phone/ Rev. 1/ SIP enabled  
SIP: Contact = sip:108@192.168.1.108:5060  
SIP: Record-Route = <sip:14083465118@192.168.1.74;user=phone;maddr=192.168.1.110;  
ForkingID=c61532549d38a692ffcae34698f21c8>  
SIP: CSeq = 101 INVITE  
SIP: Content-Type = application/sdp  
SIP: Content-Length = 220  
SIP:  
SDP: ----- SDP Header -----  
SDP:  
SDP: v(sdp protocol version) =0  
SDP: o(owner/creator and session identifier) =CiscoSystemsSIP-IPPhone-UserAgent 4439 20135 IN IP4 192.168.1.108  
  
SDP: s(session name) =SIP Call  
SDP: c(connection information) =IN IP4 192.168.1.108  
SDP: t(<start time> <stop time>) =0 0  
SDP: m(media name and transport address) =audio 18692 RTP/AVP 0 101  
SDP: a(media attributes) =rtmpmap:0 pcmu/8000  
SDP: a(media attributes) =rtmpmap:101 telephone-event/8000  
SDP: a(media attributes) =fmtp:101 0-11  
SDP:

----- Frame 130 -----

SIP: ----- Session Initiation Protocol -----  
SIP:  
**SIP: R SIP/2.0 Response Status Code=200(Success) Response-Phrase=OK**  
SIP: Via = SIP/2.0/UDP 192.168.1.74:58440  
SIP: From = "14083465118" <sip:14083465118@192.168.1.74>  
SIP: To = <sip:108@192.168.1.110;user=phone;phone-context=national>;tag=f26b030083a1d410-0  
SIP: Call-ID = 4E43092B-AD0ECA15-0-F13A2A8@192.168.1.74  
SIP: Server = Cisco IP Phone/ Rev. 1/ SIP enabled  
SIP: Contact = sip:108@192.168.1.108:5060  
SIP: Record-Route = <sip:108@192.168.1.108;maddr=192.168.1.110;  
ForkingID=c61532549d38a692ffcae34698f21c8>  
SIP: CSeq = 101 INVITE  
SIP: Content-Type = application/sdp  
SIP: Content-Length = 220  
SIP:  
SDP: ----- SDP Header -----  
SDP:  
SDP: v(sdp protocol version) =0  
SDP: o(owner/creator and session identifier) =CiscoSystemsSIP-IPPhone-UserAgent 4439 20135 IN IP4 192.168.1.108  
SDP: s(session name) =SIP Call  
SDP: c(connection information) =IN IP4 192.168.1.108  
SDP: t(<start time> <stop time>) =0 0  
SDP: m(media name and transport address) =audio 18692 RTP/AVP 0 101  
SDP: a(media attributes) =rtmpmap:0 pcmu/8000  
SDP: a(media attributes) =rtmpmap:101 telephone-event/8000  
SDP: a(media attributes) =fmtp:101 0-11  
SDP:

----- Frame 131 -----

SIP: ----- Session Initiation Protocol -----  
SIP:  
**SIP: ACK sip =108@192.168.1.108:5060 SIP/2.0**  
SIP: Via = SIP/2.0/UDP 192.168.1.74:58440  
SIP: From = "14083465118" <sip:14083465118@192.168.1.74>  
SIP: To = <sip:108@192.168.1.110; user=phone; phone-context=national>; tag=f26b030083a1d410-0  
SIP: Date = Wed, 03 Mar 1993 22:15:45 UTC  
SIP: Call-ID = 4E43092B-AD0ECA15-0-F13A2A8@192.168.1.74  
SIP: Route = <sip:108@192.168.1.108:5060>  
SIP: Max-Forwards = 6  
SIP: Content-Type = application/sdp  
SIP: Content-Length = 134  
SIP: CSeq = 101 ACK

```

SIP:
SDP: ----- SDP Header -----
SDP:
SDP: v(sdp protocol version) =0
SDP: o(owner/creator and session identifier) =CiscoSystemsSIP-GW-UserAgent 7802 3827 IN IP4 192.168.1.74
SDP: s(session name) =SIP Call
SDP: c(connection information) =IN IP4 192.168.1.74
SDP: t(<start time> <stop time>) =0 0
SDP: m(media name and transport address) =audio 20080 RTP/AVP 0
SDP:

```

Once the connection has been established, audio packets can flow between the two endpoints. The audio information is encapsulated using an Internet Protocol (IP) header, a User Datagram Protocol (UDP) header, and finally a Real-time Transport Protocol (RTP) header, as illustrated in Figure 5.

From Trace 3, note that within the IP header, the source and destination IP addresses indicate the end-to-end nature of this communication, without the ongoing involvement of the proxy server (review Figure 4B). The RTP header indicates a number of parameters, including the RTP version (2, from RFC 1889), a timestamp for this sample (668592, indicating 83,574.000 milliseconds), and the Payload Type (0, indicating PCM mu-Law encoded audio). Figure 5 illustrates each of these fields, which can be compared with the Sniffer details shown in Trace 3.

### TRACE 3. RTP Packet with G.711 Audio Payload Details

```

----- Frame 138 -----
DLC: ----- DLC Header -----
DLC:
DLC: Frame 138 arrived at 14:02:02.4409; frame size is 214 (00D6 hex) bytes.
DLC: Destination = Station Cisco C5D202
DLC: Source = Station Cisco F2411D
DLC: Ethertype = 0800 (IP)
DLC:
IP: ----- IP Header -----
IP:
IP: Version = 4, header length = 20 bytes
IP: Type of service = A0
IP: 101. .... = CRITIC/ECP
IP: ...0 ... = normal delay
IP: .... 0... = normal throughput
IP: .... .0.. = normal reliability
IP: .... .0. = ECT bit - transport protocol will ignore the CE bit
IP: .... ..0 = CE bit - no congestion
IP: Total length = 200 bytes
IP: Identification = 14029
IP: Flags = 0X
IP: .0. .... = may fragment
IP: .0. .... = last fragment
IP: Fragment offset = 0 bytes
IP: Time to live = 64 seconds/hops
IP: Protocol = 17 (UDP)
IP: Header checksum = BEB1 (correct)
IP: Source address = [192.168.1.108]
IP: Destination address = [192.168.1.74]
IP: No options
IP:
UDP: ----- UDP Header -----

```

```

UDP:
UDP: Source port   = 18692
UDP: Destination port = 20080
UDP: Length       = 180
UDP: No checksum
UDP: [172 byte(s) of data]
UDP:
RTP: ----- Real-Time Transport Protocol -----
RTP:
RTP: Ver, Pad, Ext, CC:   = 80
RTP:      10.. .... = Version = 2 (RFC 1889)
RTP:      ..0. .... = Padding = 0 (Zero bytes of Padding at the End)
RTP:      ...0 .... = Header Extension Bit = 0, (No Header Extension after Fixed Header)
RTP:      .... 0000 = Contributor Count = 0
RTP: Marker, Payload Type: = 00
RTP:      0... .... = Marker 0
RTP:      .000 0000 = Payload Type 0 (PCMU audio)
RTP: Sequence Number     = 4069
RTP: Time Stamp        = 668592 (83574.000 ms)
RTP: SSRC                = 490861163
RTP:
RTP: Payload Type = 0 (PCMU audio)
RTP:

```

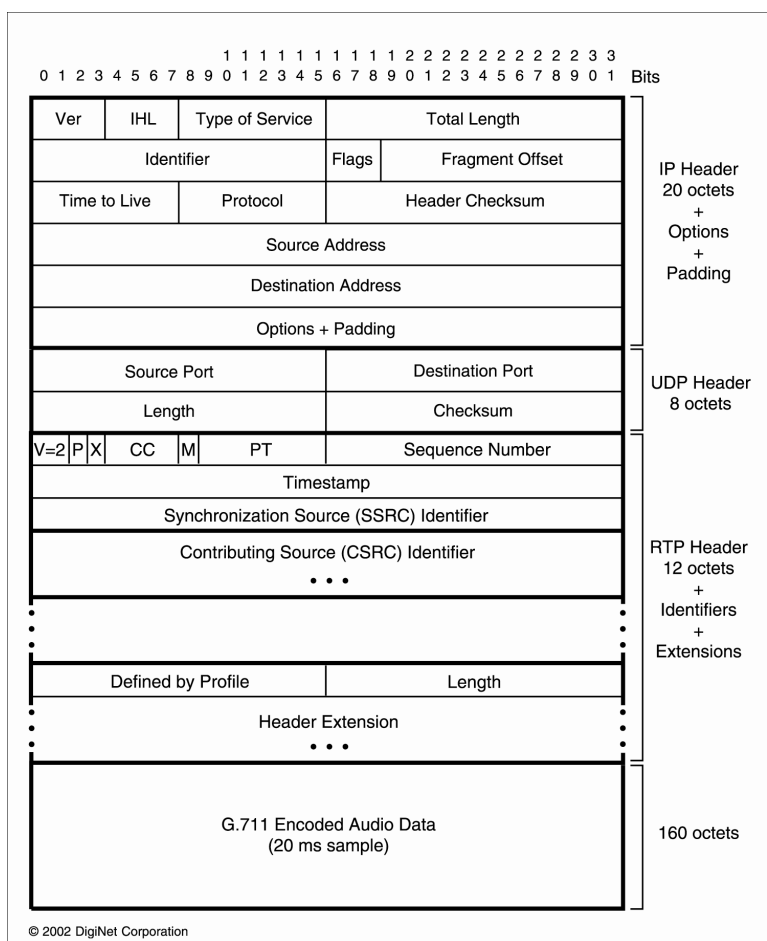


FIGURE 5. Voice over IP Packet Format with G.711 Encoded Audio Data

When the two parties have completed their conversation, the call is terminated, beginning with the BYE message sent from the Cisco Telephone in Frame 226 (review Figure 4B).

Note in Trace 4 that the same Call ID that was used in the INVITE message is referenced in the BYE message (4E43092B-AD0ECA15-0-F13A2A8@192.168.1.74). Two other confirmation messages, TRYING (Frame 227) and BYE (Frame 228), are passed between the proxy server and the end stations, followed by an end-to-end OK as the final confirmation in Frame 229. The previously mentioned Call ID is contained in these messages as well.

#### TRACE 4. SIP Call Disconnect Message Details

----- Frame 226 -----

```
SIP: ----- Session Initiation Protocol -----
SIP:
SIP: BYE sip =14083465118@192.168.1.74;user=phone SIP/2.0
SIP: Via = SIP/2.0/UDP 192.168.1.108:5060
SIP: From = <sip:108@192.168.1.110;user=phone;phone-context=national>;tag=f26b030083a1d410-0
SIP: To = "14083465118" <sip:14083465118@192.168.1.74>
SIP: Call-ID = 4E43092B-AD0ECA15-0-F13A2A8@192.168.1.74
SIP: User-Agent = Cisco IP Phone/ Rev. 1/ SIP enabled
SIP: CSeq = 101 BYE
SIP: Route = <sip:14083465118@192.168.1.74:5060;user=phone>
SIP: Content-Length = 0
SIP:
```

----- Frame 227 -----

```
SIP: ----- Session Initiation Protocol -----
SIP:
SIP: R SIP/2.0 Response Status Code=100(Informational) Response-Phrase=Trying
SIP: Via = SIP/2.0/UDP 192.168.1.108:5060
SIP: From = <sip>
SIP:
```

----- Frame 228 -----

```
SIP: ----- Session Initiation Protocol -----
SIP:
SIP: BYE sip =14083465118@192.168.1.74;user=phone SIP/2.0
SIP: Via = SIP/2.0/UDP 192.168.1.110;branch=f059ddc9dc161e2c7cd53bb1fc4d45a-
a5ca34313ad49789bf3de2774b2f7e7
SIP: Via = SIP/2.0/UDP 192.168.1.108:5060
SIP: From = <sip:108@192.168.1.110;user=phone;phone-context=national>;tag=f26b030083a1d410-0
SIP: To = "14083465118" <sip:14083465118@192.168.1.74>
SIP: Call-ID = 4E43092B-AD0ECA15-0-F13A2A8@192.168.1.74
SIP: User-Agent = Cisco IP Phone/ Rev. 1/ SIP enabled
SIP: CSeq = 101 BYE
SIP: Content-Length = 0
SIP:
```

----- Frame 229 -----

```
SIP: ----- Session Initiation Protocol -----
SIP:
SIP: R SIP/2.0 Response Status Code=200(Success) Response-Phrase=OK
SIP: Via = SIP/2.0/UDP 192.168.1.110;branch=f059ddc9dc161e2c7cd53bb1fc4d45a-
```

```
a5ca34313ad49789bf3de2774b2f7e7,  
SIP/2.0/UDP 192.168.1.108:5060  
SIP: From = <sip:108@192.168.1.110;user=phone;phone-context=national>;tag=f26b030083a1d410-0  
SIP: To = "14083465118" <sip:14083465118@192.168.1.74>  
SIP: Date = Wed, 03 Mar 1993 22:15:56 UTC  
SIP: Call-ID = 4E43092B-AD0ECA15-0-F13A2A8@192.168.1.74  
SIP: Server = Cisco VoIP Gateway/ IOS 12.x/ SIP enabled  
SIP: Content-Length = 0  
SIP: CSeq = 101 BYE  
SIP:
```

In summary, connections using SIP signaling are not as complex as those using the ITU-T H.323 protocol suite. To compare these two protocols, review the H.323 example described in Volume 4 of this series, noting the number of messages required in each case.

## 5. Looking Ahead

This is the fifth of six technical briefs on Converged Networks sponsored by Sniffer Technologies. Titles of current and future volumes in the series include:

- 1. Introduction to Converged Networking:** A description of concepts and challenges of converged networks, including business, technical, and operational issues.
- 2. Protocols for the Converged Network:** A look at the components of the converged network, the ITU-T and IETF multimedia protocol suites, and the protocols required by each component.
- 3. Implementing the Voice over IP Network:** Issues to consider before you jump in, including existing network utilization, planning for new applications, network design, and interoperability testing.
- 4. Managing Call Flows Using H.323:** The operation of the H.323 family of multimedia protocols, illustrated with case studies and output from the *Sniffer* protocol analyzer that show converged network operation from the H.323 perspective.
- 6. Supporting the Converged Network:** This concluding paper will deal with ongoing support requirements, including: traffic prioritization, WAN bandwidth optimization, and quality of service optimization.

## 6. Acronyms and Abbreviations

CCITT	Consultative Committee for International Telephony and Telegraphy
CON	Connection-oriented network service
CNLS	Connectionless network service
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
ENUM	Electronic Numbers
ETSI	European Telecommunications Standards Institute
HTTP	Hypertext Transfer Protocol
IETF	Internet Engineering Task Force
IP	Internet Protocol
ISDN	Integrated Services Digital Network
ITSP	Internet Telephony Service Provider
ITU-T	International Telecommunication Union — Telecommunications Standards Sector
LAN	Local Area Network
MGCP	Media Gateway Control Protocol
MOS	Mean Opinion Score
PBX	Private Branch Exchange
PDA	Personal Digital Assistant
PESQ	Perceptual Evaluation of Speech Quality
PSQM	Perceptual Speech Quality Measurement
PSTN	Public Switched Telephone Network
QoS	Quality of Service
RAS	Registration, Admission, and Status
RSVP	Resource Reservation Protocol
RTCP	Real-time Control Protocol
RTP	Real-time Transport Protocol
RTSP	Real-time Streaming Protocol
SAP	Session Announcement Protocol
SCCP	Skinny Client Control Protocol
SCTP	Stream Control Transmission Protocol
SDP	Session Description Protocol
SIP	Session Initiation Protocol
TCP	Transmission Control Protocol
TIPHON	Telecommunications and Internet Protocol Harmonization Over Networks
VoIP	Voice over Internet Protocol
WAN	Wide Area Network

## 7. About the Author and Sponsor

Mark A. Miller, P.E., is President of DigiNet Corporation, a Denver-based consulting engineering firm providing services in internetwork design, strategic planning, network management, and new product development. Mr. Miller is the author of nineteen books on network analysis, design, and management. His latest book is titled *Voice over IP Technologies, Strategies for the Converged Enterprise*, published in 2002 by M&T Books, Inc., a division of John Wiley (Indianapolis, Indiana). He is a frequent presenter at industry events and has taught at the ComNet, CT Expo, Internet Telecom Expo, Network+Interop, Comdex, and other conferences. He holds B.S. and M.S. degrees in electrical engineering, and is a registered professional engineer in four states. For more information, DigiNet Corporation may be reached at 303-682-5244 or on the Internet at <http://www.dignet.com>.

Sniffer Technologies, a division of Network Associates, is a leading provider of network and application management solutions designed to ensure e-business uptime. Supporting one of the widest ranges of network topologies in the industry, the Sniffer Total Network Visibility (TNV) suite is an integrated solution enabling enterprises and service providers to cost-effectively keep their networks and applications up and running at peak performance. As one of the most trusted solutions for monitoring, troubleshooting, reporting, and proactively managing network availability and performance, the Sniffer TNV suite meets the demanding 24x7 availability requirements of e-business Web sites, Internet applications, converged voice, video, and data networks, and high speed switched and optical networks. For more information, Sniffer Technologies can be reached on the Internet at <http://www.sniffer.com>.

With headquarters in Santa Clara, CA, Network Associates, Inc. is a leading supplier of network security and availability solutions. Network Associates is comprised of three product groups: McAfee, delivering world class anti-virus and security products; Sniffer, a leader in network availability and system security; and Magic, providing Web-based service desk solutions. For more information, Network Associates can be reached at 972-308-9960 or on the Internet at <http://www.nai.com>.

### Copyright

This paper is copyright © 2002 DigiNet Corporation. All rights reserved.

### Limit of Liability/Disclaimer of Warranty

Information contained in this work has been obtained by the author and sponsor from sources believed to be reliable. However, neither the author nor the sponsors guarantee the accuracy or completeness published herein, and neither the author nor the sponsor shall be responsible for any errors, omissions, or damages arising out of the use of this information. This work is published with the understanding that the author and sponsor are supplying information but are not attempting to render engineering or other professional services. If such services are required, the assistance of an appropriate professional should be sought.

### Trademarks

DigiNet is a registered trademark of Digital Network Corporation.

Network Associates, Sniffer, Total Network Visibility, TNV, McAfee, and Magic Solutions are registered trademarks of Network Associates, Inc. and/or its affiliates in the United States and/or other countries.

Cisco is a trademark of Cisco Systems, Inc. All other registered and unregistered trademarks in this document are the sole property of their respective owners.