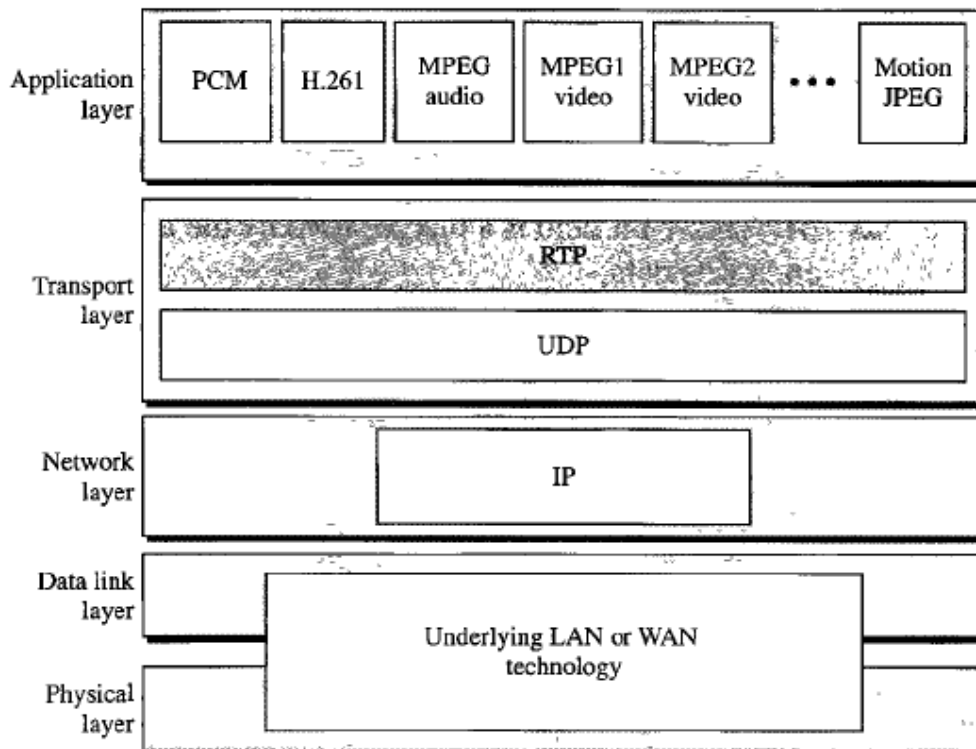


RTP

Real-time Transport Protocol (RTP) is the protocol designed to handle real-time traffic on the Internet. RTP does not have a delivery mechanism (multicasting, port numbers, and so on); it must be used with UDP. RTP stands between UDP and the application program. The main contributions of RTP are time-stamping, sequencing, and mixing facilities. Figure 1 shows the position of RTP in the protocol suite.



RTP Packet Format

Figure 2 shows the format of the RTP packet header. The format is very simple and general enough to cover all real-time applications. An application that needs more information adds it to the beginning of its payload. A description of each field follows.

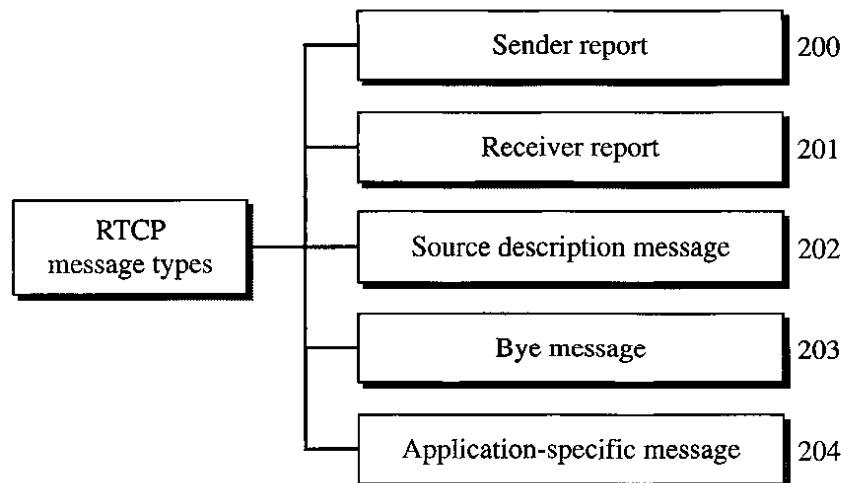
Ver	P	X	Contr. count	M	Payload type	Sequence number
Time stamp						
Synchronization source identifier						
Contributor identifier						
⋮						
Contributor identifier						

UDPPort

Although RTP is itself a transport layer protocol, the RTP packet is not encapsulated directly in an IP datagram. Instead, RTP is treated as an application program and is encapsulated in a UDP user datagram. However, unlike other application programs, no well-known port is assigned to RTP. The port can be selected on demand with only one restriction: The port number must be an even number. The next number (an odd number) is used by the companion of RTP, Real-time Transport Control Protocol (RTCP).

RTCP

RTP allows only one type of message, one that carries data from the source to the destination. In many cases, there is a need for other messages in a session. These messages control the flow and quality of data and allow the recipient to send feedback to the source or sources. Real-time Transport Control Protocol (RTCP) is a protocol designed for this purpose. RTCP has five types of messages, as shown in Figure 3. The number next to each box defines the type of the message



Sender Report

The sender report is sent periodically by the active senders in a conference to report transmission and reception statistics for all RTP packets sent during the interval. The sender report includes an absolute timestamp, which is the number of seconds elapsed since midnight on January 1, 1970. The absolute timestamp allows the receiver to synchronize different RTP messages. It is particularly important when both audio and video are transmitted (audio and video transmissions use separate relative timestamps).

Receiver Report

The receiver report is for passive participants, those that do not send RTP packets. The report informs the sender and other receivers about the quality of service.

Source Description Message

The source periodically sends a source description message to give additional information about itself. This information can be the name, e-mail address, telephone number, and address of the owner or controller of the source.

Bye Message

A source sends a bye message to shut down a stream. It allows the source to announce that it is leaving the conference. Although other sources can detect the absence of a source, this message is a direct announcement. It is also very useful to a mixer.

Application-Specific Message

The application-specific message is a packet for an application that wants to use new applications (not defined in the standard). It allows the definition of a new message type.

UDPPort

RTCP, like RTP, does not use a well-known UDP port. It uses a temporary port. The UDP port chosen must be the number immediately following the UDP port selected for RTP. It must be an odd-numbered port.

VOICE OVER IP

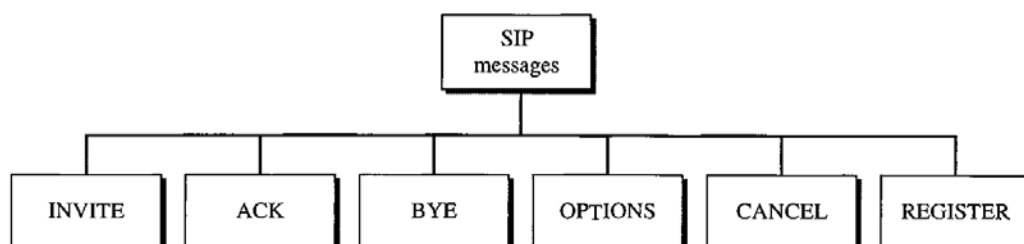
Let us concentrate on one real-time interactive audio/video application: voice over IP, or Internet telephony. The idea is to use the Internet as a telephone network with some additional capabilities. Instead of communicating over a circuit-switched network, this application allows communication between two parties over the packet-switched Internet. Two protocols have been designed to handle this type of communication: SIP and H.323. We briefly discuss both.

SIP

The Session Initiation Protocol (SIP) was designed by IETF. It is an application layer protocol that establishes, manages, and terminates a multimedia session (call). It can be used to create two-party, multiparty, or multicast sessions. SIP is designed to be independent of the underlying transport layer; it can run on UDP, TCP, or SCTP.

Messages

SIP is a text-based protocol, as is HTTP. SIP, like HTTP, uses messages. Six messages are defined, as shown in Figure 4.

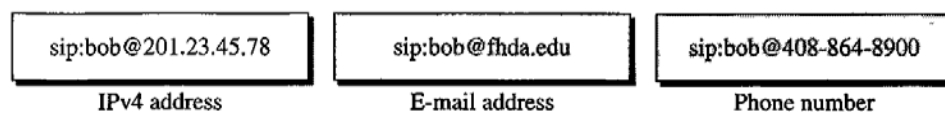


Each message has a header and a body. The header consists of several lines that describe the structure of the message, caller's capability, media type, and so on. We give a brief description of each message. Then we show their applications in a simple session.

The caller initializes a session with the INVITE message. After the callee answers the call, the caller sends an ACK message for confirmation. The BYE message terminates a session. The OPTIONS message queries a machine about its capabilities. The CANCEL message cancels an already started initialization process. The REGISTER message makes a connection when the callee is not available.

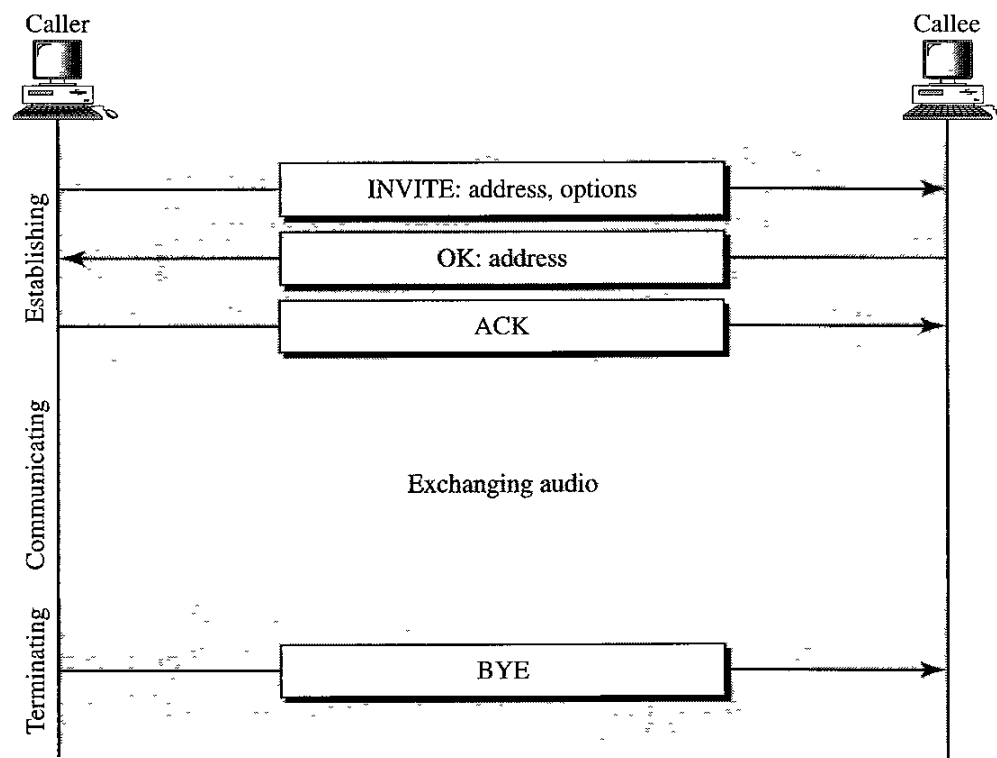
Addresses

In a regular telephone communication, a telephone number identifies the sender, and another telephone number identifies the receiver. SIP is very flexible. In SIP, an e-mail address, an IP address, a telephone number, and other types of addresses can be used to identify the sender and receiver. However, the address needs to be in SIP format (also called *scheme*). Figure 5 shows some common formats.



Simple Session

A simple session using SIP consists of three modules: establishing, communicating, and terminating. Figure 29.23 shows a simple session using SIP.



Establishing a Session Establishing a session in SIP requires a three-way handshake. The caller sends an INVITE message, using UDP, TCP, or SCTP to begin the communication.

If the callee is willing to start the session, she sends a reply message. To confirm that a reply code has been received, the caller sends an ACK message.

Communicating After the session has been established, the caller and the callee can communicate by using two temporary ports.

Terminating the Session The session can be terminated with a BYE message sent by either party.

Tracking the Callee

What happens if the callee is not sitting at her terminal? She may be away from her system or at another terminal. She may not even have a fixed IP address if DHCP is being used. SIP has a mechanism (similar to one in DNS) that finds the IP address of the terminal at which the callee is sitting. To perform this tracking, SIP uses the concept of registration.

SIP defines some servers as registrars. At any moment a user is registered with at least one registrar server; this server knows the IP address of the callee.

When a caller needs to communicate with the callee, the caller can use the e-mail address instead of the IP address in the INVITE message. The message goes to a proxy server. The proxy server sends a lookup message (not part of SIP) to some registrar server that has registered the callee. When the proxy server receives a reply message from the registrar server, the proxy server takes the caller's INVITE message and inserts the newly discovered IP address of the callee. This message is then sent to the callee. Figure 6 shows the process.

