

# Voice Over IP Implementation Using Session Initiation and Media Gateway Control Protocols

Ossi Kauranen and Jari Kellokoski

*Lappeenranta University of Technology, P.O. Box 20,53851 Lappeenranta, Finland*

{ossi.kauranen, jari.kellokoski}@lut.fi

**Key words:** Media Gateway, MGCP, SIP, RTP, VoIP

**Abstract:** This paper investigates the feasibility of using several different protocols for building VoIP gateway. The protocols that were used in the implementation were Session Initiation Protocol (SIP), Media Gateway Control Protocol (MGCP), Real-time Transfer Protocol (RTP) and Session Description Protocol (SDP). Implementation has shown that these protocols are suitable for building VoIP gateways and services. They also offer a good foundation for different kinds of multimedia services in the future.

These protocols are also suitable for similar kinds of implementations in the wireless mobile world. Personal Digital Assistants (PDAs) are already capable of handling VoIP traffic in, for example, companies Local Area Networks. A metropolitan area range is out of the question at the moment, but there are projects that will try to cover the whole city area. When 3G and 4G networks are developed fully, mobile VoIP services will be available for all users.

## 1. INTRODUCTION

With the introduction of Public Switched Telephone Network / IP gateway users finally got the possibility of completing calls over IP networks. The authors of this paper anticipate that more and more of the Internet's bandwidth will be consumed by voice and video applications. The

in the implementation of a VoIP gateway in the Linux environment. Candidate protocols are: Session Initiation Protocol, Media Gateway Control Protocol, Real-time Transfer Protocol and Session Description Protocol. The main purpose of this paper is to give brief overview of the protocols and to describe how these protocols could be used in the Linux environment to build VoIP gateway. Finally we will suggest how we might further develop and improve gateway services with other advanced features such as Resource Reservation Protocol, Quality of Service and Service Discovery.

## **2. PROTOCOLS**

### **2.1 Session Initiation Protocol**

Handley et al [1] define The Session Initiation Protocol (SIP) as an application-layer control and signalling protocol for creating, modifying and terminating sessions with one or more participants. Applications can be Internet multimedia conferences, Internet telephone calls and even multimedia distribution. Members in these sessions communicate via multicast or unicast relations. SIP invitations are used to create sessions and they carry session descriptions, which allow participants to agree on a set of media types. Session can also be invited without any resource reservation. SIP supports user mobility by proxying and redirecting requests to the user's current location. This means that user's must register their current location. However, all this provides the end user with the ability to originate and receive calls and access subscribed telecommunication services from any terminal in any location, and provides the network with the ability to identify users as they move. SIP can work with User Datagram Protocol (UDP) as well as Transmission Control Protocol (TCP) because it has its own reliability system. Invitations can be used to initiate sessions and to invite members to existing sessions. SIP is text based, making implementation easy, extensible and flexible.

#### **2.1.1 SIP components**

The basic components of the SIP system are the User Agent (UA) and the network server. The User Agent is a combination of two elements: User Agent Client (UAC) and User Agent Server (UAS) (referred to in Figure 1 as simply SIP client). The UAC initiates call requests while UAS receives requests and responses of calls, which accept, reject or redirect the requests. The network servers are a SIP proxy server and a redirect server. The SIP proxy forwards each request to the next server after it has decided where to

forward the request. The next server can be any kind of SIP server and the forwarding proxy doesn't have to know the type of the next server, so. The request might travel through several servers until it reaches the UAS. In response to that request the path will be followed in reverse order. The redirect server does not forward any request to the next server. Instead, it sends redirect responses back to the client, where it states the address of the next server to be contacted.

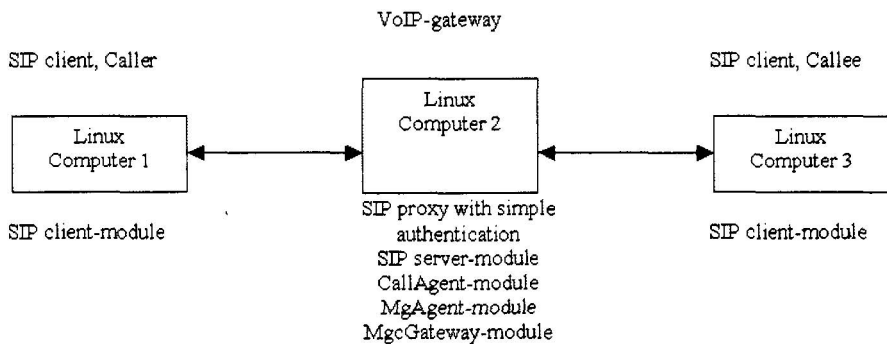


Figure 1. Our VoIP architecture

### 2.1.2 SIP vs H.323, why SIP

Internet telephony services need a set of control protocols for establishing connections, exchanging capabilities, and controlling conferences. Two standards are used to meet the challenge. One is ITU Recommendation H.323 and the other is IETF Session Initiation Protocol. While we do not intend to discuss H.323 in detail, we would nevertheless like to discuss some of the reasons why we are using SIP instead of H.323. These two standards have different kind of approach to the problem domain. While the H.323 presents a more traditional solution based on circuit-switching like the approach the ISDN protocol and previous H-series recommendations. The approach of SIP, on the other hand, is simpler, and based on the HTTP protocol, reusing many of its encoding, header files, error codes and authentication methods. Note, that in both cases the actual multimedia will be most likely transported via Real Time Protocol (RTP), so signaling does not have an affect on Internet telephony QoS.

H.323 is quite a complex protocol, using hundreds of elements and protocol components, which it defines, and there is no really good separation of these components. Even in simple cases, H.323 needs interactions between these elements and components, whereas in SIP there are only 37

headers and small number of values that contain more information. SIP encodes messages as text, similar to HTTP and RTP, leading to easy parsing. SIP is quite a simple protocol, implementing basic call signaling, user location and registration while all more advanced features of signaling (QoS, service discovery, session content description) are separate protocols. As in H.323, these features are in the protocol, making replacement of these features with new ones harder than in SIP. H.323 also requires a reliable transport protocol while in SIP the protocol can run on top of any protocol offering reliable or unreliable byte stream or datagram services without any changes to the protocol. Also, extending video or audio codecs is more complicated in H.323, where all codecs must be centrally registered and standardized; by contrast, SIP uses the Session Description Protocol (SDP) to tell the codecs supported by an endpoint in a session. Codecs are identified by string names, so SIP can work with any codec. In the services both protocols can offer wide range of similar services, although there are some differences between the mechanism for implementing these services. The basic reason for the use of SIP is the lower complexity, reasonable modularity, good extensibility and fast phase of development taking place in Internet telephony services.

## **2.2 Media gateway control protocol**

Media Gateway Control Protocol is defined by the Internet Engineering Task Force's Media Gateway Control workgroup (also called megaco). Arango et al [2] describe a telephony gateway as follows: "A telephony gateway is a network element that provides conversion between the audio signals carried on telephone circuits and data packets carried over the Internet or over other packet networks." Media Gateway Control Protocol is a protocol that controls the media gateways. MGCP assumes that the call control intelligence is outside the gateway and handled by external elements (see Figure 2).

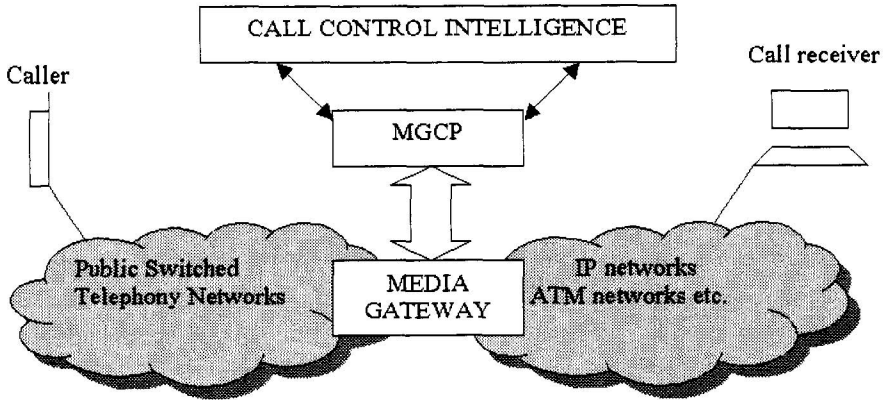


Figure 2. Overview of general VoIP gateway architecture

The main purpose of the MGCP protocol is to send/receive commands to/from the Media Gateway. This is achieved with only 8 message types. External call control intelligence will provide the needed information to the MGCP so that it can prepare the Media Gateway to know what kinds of attributes or characteristics the parties that are involved in this particular call have. These attributes can be available bandwidth for caller or call receiver, for example, or network types, IPv4 or IPv6 addresses and so on.

MGCP is an essential protocol when building a “passage” between different networks and signaling systems. MGCP assures that end-users do not have to concern themselves about the different network characteristics.

## 2.3 Real-time transfer protocol

RTP stands for Real-time Transfer Protocol. RTP and the Real-time Control Protocol are defined by the IETF’s Audio-Video Transport Working Group, Schulzrinne et al [3]. RTP provides functionalities for transporting real-time data streams such as voice, audio or video over unicast or multicast networks. RTP is designed for multi-participant multimedia conferences. RTP doesn’t reserve resources for a given connection nor does it guarantee any Quality of Service features. These functionalities are left for more appropriate protocols. RTCP provides minimal control and identification functionalities.

Because RTP and RTCP are network independent protocols they are very suitable for transmitting voice over different networks in VoIP solutions. Another big bonus comes from their main purpose: multi-participant multimedia conferences. In university environments, for example, where the available bandwidth might easily be 10 - 100 Mbps/second or more, there

would be a wide range of possible applications, including videoconferences, virtual teaching, etc.

## 2.4 Session description protocol

Session Description protocol is a product of the IETF's Multiparty Multimedia Session Control working group, Handley et al [4]. SDP provides information about multimedia media streams so that a recipient of an SDP message can participate in these "sessions". SDP information is carried in both SIP and MGCP messages. This guarantees co-operation of the protocols. SDP forms the link between SIP and MGCP in the sense that it provides information about the end-users to the MGCP. The standard form of session information is very useful for VoIP solutions. This knowledge makes the whole system very adaptable.

## 3. IMPLEMENTATION

Our implementation tries to utilize the basic idea of VoIP service with both SIP and MGCP protocols, thus using. The SIP protocol in signaling, while the MGCP protocol controls the gateway. The multimedia itself is transferred using the RTP protocol. One example of the procedure with VoIP calls is following in the SIP and MGCP protocols (Figure 3). A caller wants to call to a callee. First, the UAC in the SIP protocol launches the invite request to the SIP Proxy server. The proxy server uses the location server to locate the callee and forwards the invitation to the callee. At this stage the MGCP uses a Createconnection message to tell the MG to create the first part of the connection. The callee's UAS replies to the proxy with reply code 100 ~ trying which is then forwarded to caller. When the callee accepts the call from the caller, the callee's UAS sends code 200 ~ OK to proxy, which forwards it to the caller. When the callee's 200 ~ OK message arrives at the MGCP it sends a second Createconnection message to the MG. Now the second part of the connection is ready. The caller acknowledges this (through the proxy) and after that both way RTP media is transferred (the actual call). When the call is ended, a BYE message is transmitted and acknowledged with code 200 ~ OK (through the proxy). When MGCP notices the BYE message it deletes one part of the connection. The other part of the connection is deleted when the ACK to the BYE message is received. When the connections are deleted the MG calculates the bytes send/received and forwards this and possibly other information to the MGCP. From these messages the Invite and the 200 ~ OK carry session description information

which is used by MGCP and RTP protocols to control the gateway and initiate the actual multimedia transfer.

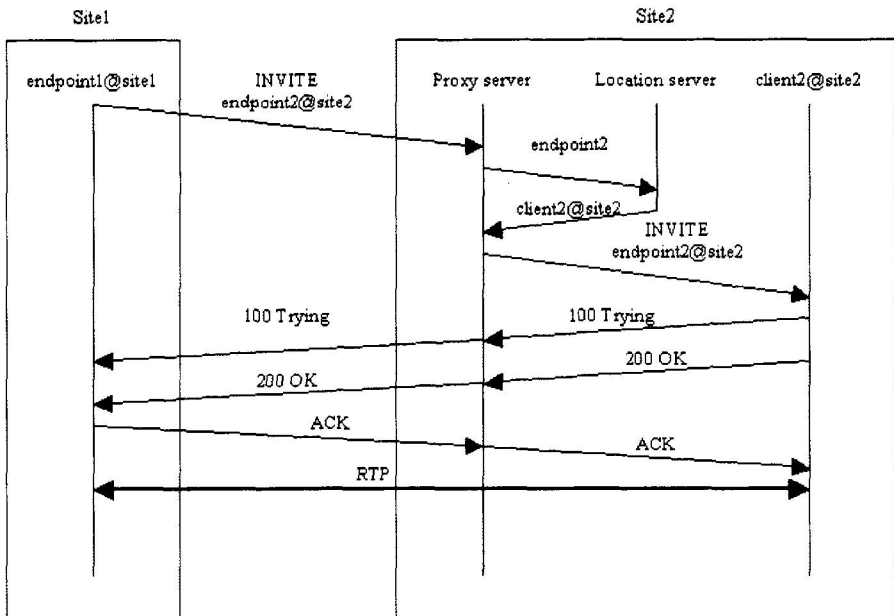


Figure 3. Simple SIP signaling example

Some basic notes about SIP matters; MG and SIP have no relationship at all. Only MGCP needs to understand both MGCP and SIP protocols. From the point of view of the SIP protocols, the MGCP side looks like a node with a large number of connections. Moreover, the MG is completely unaware that the call between end users is established via SIP.

The implementation has proven itself. Media Gateway can be used to transfer video, voice and other real-time streams while SIP protocol provides easy and simple signalling. As a bonus MGCP provides the means to gather information about the traffic that has transferred through the gateway. This information can be used for, for example, billing purposes etc.

#### 4. CONCLUSIONS AND FUTURE WORK

Given the importance of telephony services, we anticipate that a major part of the Internet bandwidth will be consumed by voice and video, i.e. RTP-based protocols. Due to its tight delay constraints, VoP streams are also likely candidates for guaranteed QoS. One question is whether we can

somehow limit the bandwidth between different users. This means that customers who pay more will have more bandwidth than others. A natural consequence of this is that we must guarantee that the user will get the bandwidth that he pays for. Reservation Protocol could give answers to this question. Unfortunately, existing RSVP protocols are rather complex and they are not 'Internet wide'. In our experience, RTP-based, MGCP and SIP protocols are well suited for further development of VoIP solutions. These rapidly developing protocols are independent and modular, and when combined with billing, service discovery and resource reservation protocols, they are well suited for future services.

## REFERENCES

- [1] Handley M., Schulzrinne H., Schooler E., Rosenberg J., 1999 IETF RFC 2543,6-8
- [2] Arango M., Dugan A., Elliott I., Huitema C., Pickett S., 1999 IETF RFC 2705, 5-6
- [3] Schulzrinne H., Casner S., Frederick R., Jacobson V., 1996 IETF RFC 1889,3-4
- [4] Handley M., Jacobson V., 1998 IETF RFC 2327, 1-4