WHAT’S BEHIND YOUR SMARTPHONE ICONS?

A brief tour of behind-the-scenes signaling for multimedia services

Harry G. Perros
Computer Science Department
NC State University,
Raleigh 27695
USA
Email: hp@ncsu.edu
Tel: 919-515-2041
**Introduction**

In our daily life we use regularly various multimedia services, such as, voice and video calling (Skype, Facetime, Google talk), video conferencing (WebEx, Telepresence), TV delivered over the Internet (IPTV), video on demand (YouTube, Netflix, Amazon, Google movies) and instant messaging (texting). We also use other apps that often include some of the above multimedia services, such as, geo-tagging, online games, and assisted living.

Have you ever wondered how apps such as Facetime, Skype, and Google talk work? What transpires from the moment you call somebody to the moment that you see and hear the other person? It all has to do with the magic of signaling! In this article, I give a brief explanation of some of behind-the-scenes signaling that make such multimedia services possible.

A multimedia service permits two devices to communicate using a combination of various media, such as, audio, video, text, still image, and animation. It is enabled using various signaling protocols, of which the most popular is the *Session Initiation Protocol* (SIP), and its extension, the *IP Multimedia Subsystem* (IMS). Loosely speaking, a signaling protocol for multimedia services is a scheme through which a communication between two users can be enabled. In its simplest form, it can locate the IP address of the person you call, alert the calling party, and make sure that the two devices can talk to each other using the same coding scheme. Of course, there are a lot more complex operations than the one described above that can be handled by a signaling protocol, such as, adding another person to the conversation, switching from audio to video, transmit a photo while you are conversing, and provide service continuity as you move from a WiFi
area a to 3G or LTE area.

SIP is a signaling protocol that is used to establish, modify, and terminate multimedia sessions. You can download a SIP client on your laptop or smartphone from one of the many SIP providers and use it to communicate with another person. Also, you can install Wireshark so that you can observe all the signaling messages exchanged between the two devices in order to enable a communication. Wireshark is a monitoring software tool that analyzes packet traffic. There are several tutorials in the Internet that will help you learn how to use wireshark, such as the one listed in the reading list.

IMS is another signaling protocol for setting up multimedia sessions. It is based on SIP but it is a lot more powerful and more complex. It provides the same features for the establishment of multimedia sessions as in SIP, but in addition it provides a platform upon which new multimedia services can be developed. IMS has been adopted by the telephone operators as the signaling protocol for the 4G phones (G stands for “generation”). 4G is also known as Long Term Evolution (LTE), which is the name of the transmission technology used to communicate with a cellular tower. Below, we first describe SIP and then IMS. For further reading, consult the list of reading list given at the end of the article.

The Session Initiation Protocol (SIP)

SIP can be used to setup a simple telephone call between two users, a multi-party call, such as a group-based audio-visual conference, or a session between a user and an application server. SIP can also be used to invite participants to an already existing session, and to add and remove multimedia streams from an existing session. It also
supports a variety of application servers, such as, presence (information regarding the
status of your friends in your buddy list) and instant messaging.

In SIP, a user is identified by a SIP address, known as the SIP Uniform Resource
Identifier (URI). It looks like an email address, and it has the form: sip:name@domain.com. For instance, let us assume that Bob Smith has a SIP account
with the SIP provider freesip under the name bsmith. Then, his SIP URI is:
sip:bsmith@freesip.com. The SIP URI is a public URI that can be publicized in
directories, and it is used to contact a SIP user. However, in order to locate a user, his
public URI has to be associated with the device that he is currently using. For instance,
let us assume that Alice Smith has a SIP URI sip:alice.smith@freesip.com, and she is
currently in her office logged into her computer pc1.company.com under the login name
alice. Then, her computer, indicated as sip:alice@pc1.company.com, is associated with
the SIP user alice.smith@freesip.com. In order to enable this association, Alice has to
register with SIP when she logs into her computer. In this way, when somebody calls
alice.smith@freesip.com, SIP knows where she is, and can route the message to her
computer.

A SIP URI may also be a telephone number, such as +1-555-555-0000. If a user
dials the telephone number +1-555-555-0000, then this is converted into the SIP URI
sip:+15555550000@domain.com;user=phone, where the user parameter indicates that the
user name field left of the @ sign is a telephone number. There are other variations of the
SIP URI that permit to indicate phone numbers. Some phones also allow the user to dial
directly an IP address. For instance dialing #152.168.1.10 places a call to the IP address
152.168.1.10.
SIP consists of User Agents (UAs) and SIP servers. A UA is a software client running on a computer, or a tablet, or a smartphone, that can initiate and answer SIP calls. It may also be a SIP phone. SIP makes use of several different servers that are used to provide support for various SIP functions, such as, user registration and user location. The proxy server is the main server in SIP and it serves a number of UAs. All SIP messages issued by a UA are forwarded to its assigned proxy server, who after some processing forwards them to their destination. In the reverse direction, all SIP messages sent to a UA are first sent to its proxy server who then forwards them to the UA. The proxy server is an important part of SIP, and a SIP operator may have multiple proxy servers depending on the number of subscribers. Another important server is the registrar. This server acts as the front end to a location service, which keeps track of the device (or devices) currently being used by a SIP user. The registrar and the proxy server are logically different entities, but they may be running in the same machine.

![Figure 1: Registrar and location server](image)

An example of the registrar and the location service is shown in figure 1. In this
example, we see that Alice registers alice@pc1.company.com with the registrar at freesip.com (step 1). The registrar forwards this information to the location server (step 2). Quite independently we see that a SIP message to alice.smith@freesip.com is sent by a UA to its SIP proxy server (step 3). The proxy server learns from the location server which device Alice is using (steps 4 and 5), and then forwards the SIP message to alice@pc1.company.com (step 6).

![SIP session setup diagram]

**Figure 2: An example of a SIP session setup**

SIP is a request-response protocol and it is based on HTTP. A SIP message can be either a request (written in capitals) or a response to a request (given by a number followed by a description of the response). An example of the message flow incurred in order to setup a SIP session between two users is given in figure 2. Alice operates a smartphone, which we will refer to as Alice’s UA. Bob has also a smartphone which we
will refer to as Bob’s UA. For simplicity we assume that both Alice’s UA and Bob’s UA are served by the same proxy server. In this example, Alice initiates a call to Bob, which causes Alice’s UA to send an INVITE request message to the proxy server. The proxy server processes the message and then it forwards it to Bob’s UA. (The proxy server finds out the device that Bob is currently using by querying the registrar, as shown above in figure 1.) The proxy server sends a 100 Trying response message to Alice’s UA to indicate that it is working on routing the INVITE request message to the destination. When Bob’s UA receives the INVITE request message, it rings to alert Bob of the incoming call, and it indicates this to the proxy server by sending a 180 Ringing response message which the proxy server forwards to Alice’s UA. When Alice’s UA receives the 180 Ringing response message, it generates a ringing or it displays a message to indicate that Bob’s phone is ringing. When Bob picks up the phone, its UA sends a 200 OK response message to the proxy server which is then forwarded to Alice’s UA. In response, Alice’s UA sends an ACK request message to Bob’s UA, and at that time Alice and Bob can begin to communicate. This is done by packetizing the audio or audio/visual data produced by Alice in IP packets, and transmitting them to Bob’s UA where the data stream is reconstructed and played out to Bob. The same happens in the opposite direction, from Bob’s UA to Alice’s UA. The session is terminated when one of the two parties hangs up. A BYE request message is sent to the other party that responds with a 200 OK message.

Several different SIP requests and responses have been created in order to handle different situations and offer different SIP-based services. The following list gives some of these SIP requests:
• REGISTER: It is used to indicate which device a SIP user is currently using. This message is forwarded to the registrar, which updates the location service.

• INVITE: It is used to establish a session between two parties, or among multiple parties.

• ACK: It is used to acknowledge that a SIP session has been setup and the media (voice, video, data) can begin to flow.

• BYE: Terminates a SIP session. For instance, when the first person hangs up in a phone call between two people, a BYE is issued to the device of the other party.

• CANCEL: Cancels a pending request.

• MESSAGE: It is used to carry an instant message (texting).

• NOTIFY: It is used to notify a UA about a particular event. For instance, a presence server uses this message to send current information about the status of the users in a buddy list.

• PUBLISH: It is used to upload information. For instance, a UA uses this message to update the status of its user (on-line, off-line, away, etc.), which appears in the buddy list of another user.

• SUBSCRIBE: This message is used to receive notifications about a particular event. For instance, it is used to receive the status of the members in a buddy list.

The IP Multimedia Subsystem (IMS) – SIP on steroids!

The IP Multimedia Subsystem (IMS) is a powerful extension of SIP and it can be used to setup audio and video calls and also enable other multimedia services, such as, presence, instant messaging, IPTV, video teleconferencing, and multimedia telephony services
(caller id, call blocking, putting someone on hold, forwarding a call to a third party, custom alerting tones). It is also used for multimedia service continuity, which refers to the capability of continuing an ongoing multimedia session as the user moves across different access networks. For instance, a user with a smartphone may initiate an audio/video call through the smartphone’s WiFi interface, but during the call the user moves away from the range of the WiFi. In this case, the continuity service will transfer the call to the cellular interface seamlessly. Unlike SIP, IMS is a complex software and no free IMS service providers are currently available from where you can download a client. If you have a facility with installing software systems, you can download and install the "open source IMS" from http://www.openimscore.org. As in SIP, you can use Wireshark to observe the signaling messages.

In IMS, a user is identified in the same way as in SIP. That is, it is associated with a SIP URI which has the format sip:name@domain.com, as explained above. Unlike in SIP, an IMS user may have several SIP URIs which can be published in web pages and business cards. For instance, a user may have a work-related SIP URI and a personal SIP URI. These SIP URIs are called public identities, and they can be used to contact a user. In addition, there is a private SIP URI associated with a user known only to the user. It is assigned by the network operator, and it is used for authentication, administration and accounting purposes.

As in SIP, IMS consists of User Equipment (UE) and servers. A UE is the term used in IMS for a device, as opposed to UA used in SIP. A UE may be a smartphone, or a tablet, or a computer, that runs an IMS client. IMS uses many servers in order to enable the different services offered by a service provider. The UEs and IMS server
communicate with each other using SIP messages.

The workhorse of IMS consists of two servers, the *Proxy Call Session Control Function* (P-CSCF) and the *Serving Call Session Control Function* (S-CSCF). As shown in figure 3, the P-CSCF is the first point of contact of a UE with IMS. It acts as a proxy server for the UEs and it forwards all the signaling messages to/from the UEs from/to the S-CSCF. The S-CSCF is the main server of IMS and it is responsible for handling user registration, connecting a UE to application servers, maintaining session states, and storing the user’s profile. There may be more than one S-CSCF depending on the number of subscribers. The Home *Subscriber Server* (HSS) is the main data storage for all subscribers and service-related data of IMS. It contains user-related information, such as, user identities, security information, location information, roaming authorization, information about which application servers can be accessed, and the S-CSCF allocated to the user.
As in SIP, a user first has to register with IMS. This is done using the SIP REGISTER message, which is forwarded to the P-PCSF, which after some processing sends it to the S-CSCF. The S-CSCF processes the request, downloads the user’s profile from the HSS, and then responds to the UE with a 200 OK.

When a UE wants to call another UE, all the signaling messages go through the S-CSCF. For instance, let us assume that UE 1 in figure 3 wants to call UE 2. UE 1 sends an INVITE message to UE 2, which is routed to the S-CSCF through P-CSCF 1. The S-CSCF looks up the current location of UE 2 in the HSS, and then forwards the message to UE 2 via its P-CSCF 2. Subsequently, a number of messages are exchanged between the two UEs all routed through the two P-CSCFs and the S-CSCF, before the two UEs can start the call.

IMS was designed so that a user can access various multimedia services, such as, instant messaging, teleconference, IPTV, video on demand, and service continuity. These services are offered by specialized application servers that are accessible via the S-CSCF, as shown in figure 3. They are SIP-based and they are developed using Java. There are several tools that can be used to develop such servers, such as, SIP Servlets and the Java Application Programming Interfaces (JAIN) Service Logic Execution Environment (SLEE). SIP-based application servers is a newer development compared to the legacy CAMEL and Parlay/OSA applications used by telephone operators. Mobicents is an open source servlets-based application development tool that runs on top of IMS. You can use it to develop new apps.

An example of a service enabled through IMS is the voice continuity service. It is provided by a service continuity server that runs on top of IMS, see figure 3. As we all
know, smartphones are equipped with a WiFi and with a cellular interface that connects us to the cellular network. These are two different interfaces through which we can communicate with the Internet and also make phone calls. For instance, through the WiFi interface we can access the Internet and also make phone calls using a VoIP service. We can also communicate with the Internet and make calls through the cellular 3G or 4G (LTE) interface. Let us assume that you are out in the street and you make a call to a friend. This call will go through the cellular interface of our smartphone. Now, while you are on the phone you walk into a café, where there is a free WiFi. Without your knowledge, the call is switched from the cellular interface to the WiFi. This is done in order to keep the cost of the call low, and also ease up the traffic between cell phones and a cellular tower. Of course, the opposite is also possible. That is, if you leave the café while still on the phone, the call will get switched to the cellular interface. A variant of this service exists, but it is not widespread yet.

The question is how is it enabled? What kind of behind-the-scene signaling takes place that enables this kind of transfer?

An example of this scenario is shown in figure 4. Smartphone UE 1 is away from a WiFi and it calls smartphone UE 2 through its LTE interface. UE 2 is currently accessible only through LTE. During the call, UE 1 moves into a WiFi zone, it senses a good WiFi signal and using IMS signaling switches the call to the WiFi interface. How is this done? When UE 1 initiates originally the call to UE 2, the IMS signaling for this call is routed through the service continuity server. We say that the call is anchored at the service continuity server. When UE 1 senses the WiFi signal and decides to switch from LTE to WiFi, it sets up a new call to UE 2. This call is also routed through the service
continuity server who realizes that UE 1 is trying to switch the call to the WiFi interface. It informs UE 2 of the IP address of the WiFi interface that UE 1 wants to use, so that UE 2 can now send all the IP packets with the audio/visual media data directly to this new IP address. Subsequently, it drops the call from UE 1 via the LTE interface. Assuming that this is done in a seamless way, neither party will become aware of the switch! There are different scenarios of service continuity, some of which are described in my book given in the reading list.

Figure 4: Switching from LTE to WiFi

Conclusions

SIP and IMS are two protocols used to enable the long-term goal of running all networking services, including voice calls, over the IP network. The Quality of Service (QoS) of a voice/video call transported over the IP network is critical to the operators and it can be guaranteed by using appropriate QoS schemes, such as MPLS with DiffServ, in the IP network. Currently, service continuity between WiFi and 3G/LTE is not always seamless, but this will improve with time. In general, the future of networking services is bright, and we should expect to see more of them in various aspects of our lives.

Reading list

• How to use Wireshark to capture, filter and inspect packets: http://www.howtogeek.com/104278/

• Open IMS core: http://www.openimscore.org

• Mobicents, http://www.mobicents.org/

• Camarillo, G. and Garcia-Martin, M. A. (2008), The 3G IP multimedia subsystem (IMS): merging the Internet and the cellular worlds, Wiley.