Mobility Support using SIP and RTP

January 22, 1999

Abstract

Enabling mobility in IP networks is an important issue for making use of the many light weight devices appearing at the market. The IP mobility support being standardized in the IETF uses tunnelling of IP packets from a Home Agent to a Foreign Agent to make the mobility transparent to the higher layer. The problem with triangular routing can be solved by using route optimization, where the sender tunnels the packets itself. There are several drawbacks with this approach, and we will in this paper propose to use mobility support in SIP and RTP where applicable, in order to support realtime communication in a more efficient way.

1 Introduction

The IP mobility support being standardized within the IETF [1] provides for *transparent mobility*, in that it hides the change of IP address when the mobile host is moving between IP subnets. This is for example needed to keep TCP connections alive. However, mobile IP is struggling with the problem of triangular routing, i.e., a packet to a mobile host travels via the home agent, whereas a packet from the mobile host is routed directly to the destination. The route optimization [2] solves this by sending binding updates to tell the sending host about the actual location of the mobile host. This solution has several problems, as will be discussed in the next section.

For realtime traffic, e.g. voice over IP, it is more common to use the Realtime Transport Protocol (RTP) [3] over UDP, and important issues are fast handoff, low latency, and - especially for wireless networks - high bandwidth utilization. Therefore, we see a need to introduce mobility awareness on a higher layer, where we can utilize knowledge about the traffic to make decisions on how to handle mobility in different situations. The Session Initiation Protocol (SIP) [4] already supports personal mobility[1], and the changes needed to support device mobility are minor. In addition, RTP is using a location-independent identifier for each media stream, which can be utilized for mobility support with practically no change to the specification. In this document, we will discuss how mobility support in SIP and RTP can improve the performance for realtime services in wireless networks, and propose an architecture for how this can be done.

Throughout this document “mobile IP” refers to IP mobility support as defined in [1].

2 IP Mobility Support

An IP address is location dependent, which means that if a host moves away from its home network, it must be assigned a new IP address that points to the new location. There are two problems with this: One is that DNS entries and other information about the host will point to the old IP address. The second problem is that existing TCP

1.“Personal mobility is the ability of end users to originate and receive calls and access subscribed telecommunication services on any terminal in any location, and the ability of the network to identify end users as they move. Personal mobility is based on the use of a unique personal identity (i.e. ‘personal number’).” [5, p. 44].

connections will break, since a TCP connection is identified by source and destination IP addresses and port numbers. The objective of mobile IP is to provide transparent support for mobility by letting a mobile host always be reachable through one address, the home address. A home agent on the home network tunnels IP packets to the current location of the mobile host.

In this section we will use a simple example to review the mechanisms of mobile IP. We use the following entity names:

- Mobile Host (MH): The host that is moving between IP networks.
- Correspondent Host (CH): The communication endpoint to the mobile host, can be either mobile or fixed.
- Home Agent (HA): A router in the mobile host's home network which can encapsulate and tunnel packets to the mobile host.
- Foreign Agent (FA): Either a router in the mobile host's visited network, or the mobile host itself. The foreign agent must be capable of decapsulating tunnelled packets.

2.1 Mobile IP

Let us consider a case where both the mobile host and correspondent host are sending realtime data (RTP streams) to each other. Moreover, we assume that the mobile host will use a foreign agent for getting its IP address and to decapsulate the incoming packets.

When using mobile IP, the following will happen: While the mobile host is still in its home network, the data is transmitted as usual, as shown in Fig. 1.

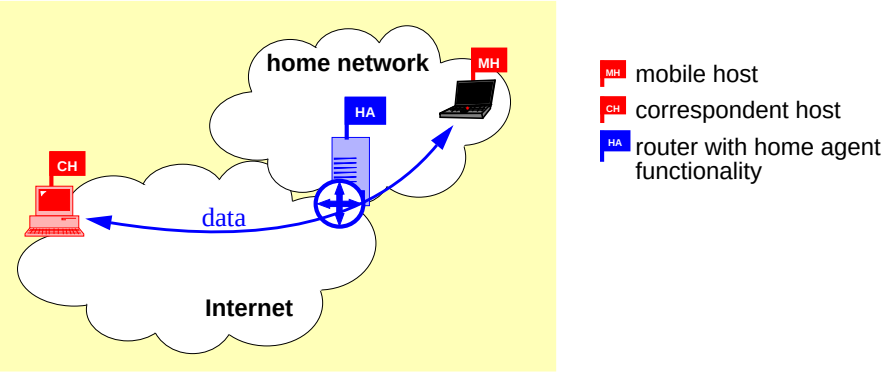

Figure 1. Mobile host at Home Network

When the mobile host moves to a new network and finds a foreign agent, it will send a registration to the home agent informing it about the address of the foreign agent. When the home agent receives packets destined to the mobile host, it will encapsulate and tunnel them to the foreign agent, which will decapsulate and send them to the mobile host (see Fig. 2).

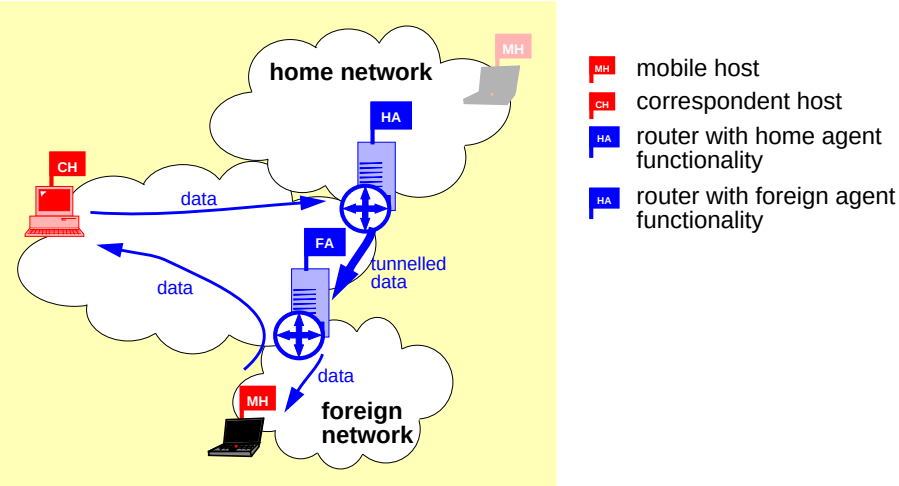

Figure 2. Mobile host moves to new network

The latency will be significantly larger when the packets are routed via the home agent instead of directly to the mobile host. The increased latency for one packet sent from the correspondent host to the mobile host will be

$$\Delta t = t_{\text{CH-HA}} + t_{\text{encapsulation}} + t_{\text{HA-FA}} + t_{\text{decapsulation}} - t_{\text{CH-FA}}$$

Measurements [9] show that mobile IP increases the latency by 45% within a campus, which can be expected to increase in a wide area network, when the distance increases between the different entities. These numbers are also highly dependent on the mobile IP implementation, and the capacity of the home agent and foreign agent. For delay sensitive traffic, this is not acceptable, because it cannot afford a higher latency in the network than what is absolutely necessary.

Moreover, the tunnelling of packets adds packetization overhead, which for IP-in-IP encapsulation [6] is typically 20 bytes (one IP header). The minimal encapsulation scheme [7] adds 12 bytes. Compare this to the packet size for an audio packet, which is around 60 bytes including IP, UDP, and RTP headers, if the coder's bitrate is 6 kbit/s.

2.2 Mobile IP with Route Optimization

In order to minimize latency, a route optimization protocol is being developed as an extension to mobile IP [2]. Using this scheme, the correspondent host will bypass the home agent and tunnel the packets itself to the foreign agent.

Whenever the home agent receives a packet that it has to tunnel to the mobile host, it will send a binding update to the correspondent host, in the form of a UDP packet sent to the well-known port 434. The correspondent host will save this information in a binding cache, and can now encapsulate and tunnel the packets to the foreign agent itself (see Fig. 3). The packets must still be tunnelled, though, in order to achieve transparent mobility.

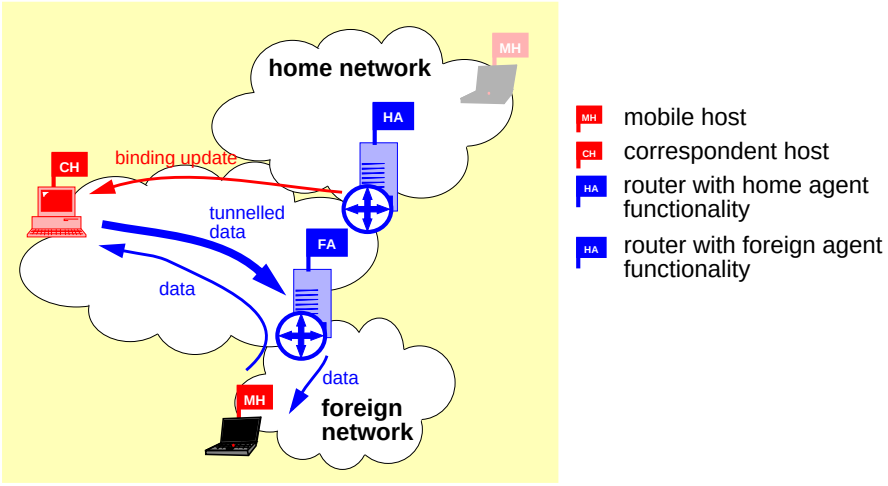

Figure 3. Mobile host moves to new network

If the mobile host moves again, it may include information about the old foreign agent in the registration request to the new foreign agent. The new foreign agent can then send a binding update to the old foreign agent so that it can forward packets to the new location of the mobile node. Furthermore, the home agent is informed of the move as defined in mobile IP. However, now the packets are being tunnelled directly from the correspondent host to the old foreign agent. The old foreign agent will tunnel the packets to the new foreign agent. When the old foreign agent receives tunnelled packets from the correspondent host, it will send a binding warning to the home agent telling it that it thinks that the correspondent host has an old binding. The home agent will then send a binding update to the correspondent host. A binding update message to the correspondent host may only be sent by the home agent due to security considerations.

There are several drawbacks with route optimization for mobile IP:

- Route optimization requires changes in the IP stack of the correspondent host, since it must be able to encapsulate IP packets, and store care-of addresses of the foreign agent or mobile host.
- The correspondent host must have a process listening on port 434 for binding update messages.
- Only the home agent may send binding updates to correspondent hosts. This means that there will be an extra delay before the correspondent host finds out where to send the packets, during which the old foreign agent must forward the packets to the correct location.
- The mobile host needs to rely on the old foreign agent forwarding packets to its new foreign agent until the correspondent host has got the binding update. There is no requirement saying that the foreign agent must do so.
- The binding warnings and updates are not compulsory, and should be used sparingly, since it can be expected that many hosts will not support the binding update function.

Because of the requirements that are put on the correspondent hosts, it cannot be expected that route optimization will be widely employed in a near future.

2.3 Security

Security is a primary issue when discussing mobile IP, since a malicious user may cause much damage redirecting traffic. IP security can be used in combination with mobile IP in order to provide authentication and/or encryption.

In some cases where the mobile host belongs to an intranet which is protected by a firewall, the firewall will not admit a packet having a source address that does not correspond to the network from which it came. To avoid this problem, a bidirectional tunnel is used: Instead of sending a regular packet with the home address as sender, the mobile host encapsulates the packet and sends it to the mobile host's home agent, which decapsulates the packet and sends it to the destination.

3 SIP Mobility Support

The Session Initiation Protocol (SIP) [4] is used for establishing and tearing down multimedia, multi-user sessions. Entities in SIP are clients, proxy servers and redirect servers. A user is addressed with an email-like address "user@host", where "user" is a user name or phone number and "host" is a domain name or numerical address. SIP defines a number of methods, listed in table 1. Responses indicate success or failure, distinguished by status codes, 1xx (100 to 199) for progress updates, 2xx for success, 3xx for redirection, and higher numbers for failure. For more information on methods and response codes, please consult [4].

Table 1: SIP Requests

Message Name	Function
INVITE	Invite user(s) to a session. The session description is contained in the body of the message, e.g. using the Session Description Protocol (SDP) [8]. The session description contains the address where the host wants to receive media streams.
ACK	Acknowledgment of an INVITE request.
BYE	Sent when a call is to be released.
OPTIONS	Query about capabilities.
CANCEL	Cancel a pending request.
REGISTER	Register with a SIP server.

SIP requests and responses are generally sent using UDP, although TCP is also supported. A typical signalling case in redirect mode is shown in Fig. 4. The "INVITE" message is received by a redirect server, which consults a location server to find out where to redirect the invitation. The function of the location server is not specified, but can be anything that can return a next hop address in the chain of finding the callee (which could be an address to another redirect server or a proxy). In Fig. 4, the location server returns the address of the callee.

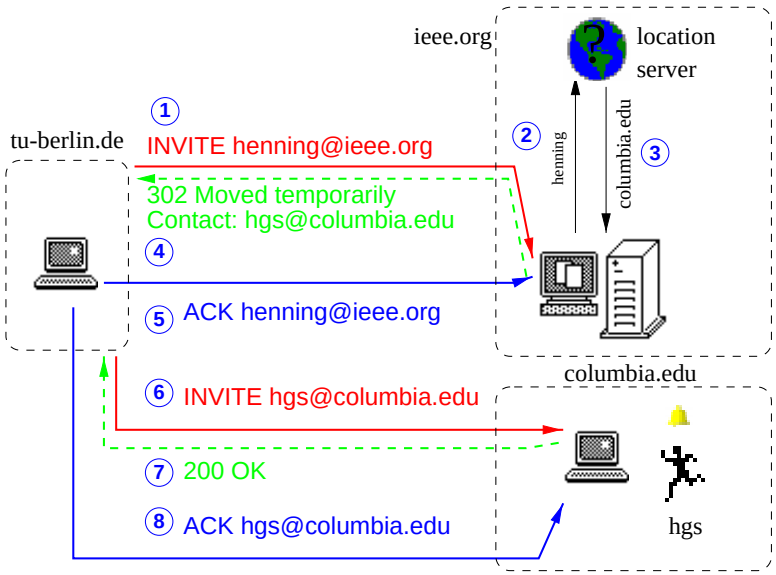

Figure 4. SIP Transaction

A proxy server would, instead of redirecting the invitation, forward it to the callee. From now on, only redirect servers will be discussed, but this does not mean that a proxy server cannot be used instead. However, the load on a redirect server can be expected to be lower since it only needs to send an answer with the user's location, instead of generating requests.

The SIP redirect server has properties resembling those of the home agent in mobile IP with route optimization, in that it tells the caller where to send the invitation. In addition, it can store preferences for the user regarding how to treat incoming requests depending on where the user is located, time of day, or the identity of the caller.

3.1 SIP Mobility Support without Mobile IP

Since most networks currently do not support mobile IP, we propose that multimedia terminals use SIP to achieve mobility. In this mode, the mobile host registers with a SIP server in its home network in order to be found. This is similar to telling the home agent that it has moved. When the correspondent host sends an INVITE to the mobile host, the redirect server has current information about the mobile host's location and redirects the INVITE there (see Fig. 5[1]).

1.For conciseness, the ACK message, needed to confirm the receipt of the response, is left out in the figures throughout the rest of this document.

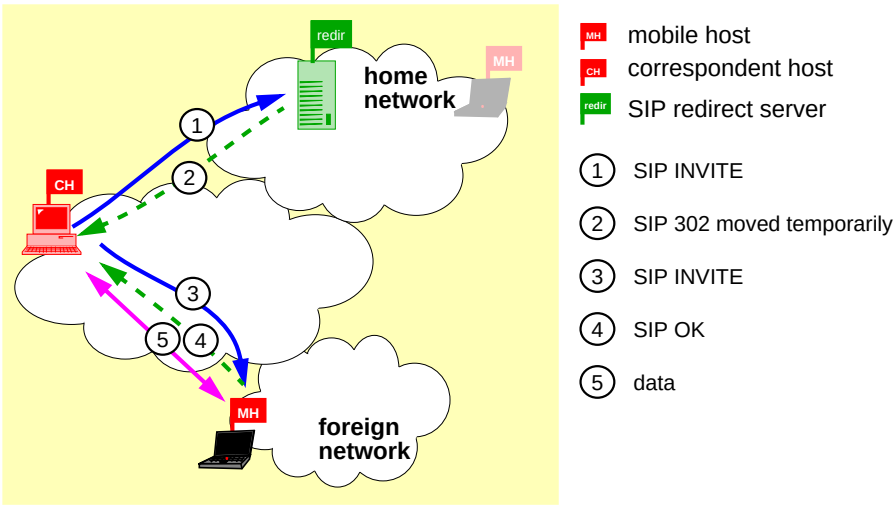

Figure 5. SIP Mobility: Invitation

If the mobile host moves during a session, it sends a new INVITE to the correspondent host using the same call identifier as in the original call setup, and puts the new IP address in the "contact" field of the SIP message, and possibly also in the session description, e.g. if it specifies transport address, see Fig. 6. It should also make a new registration at the SIP server.

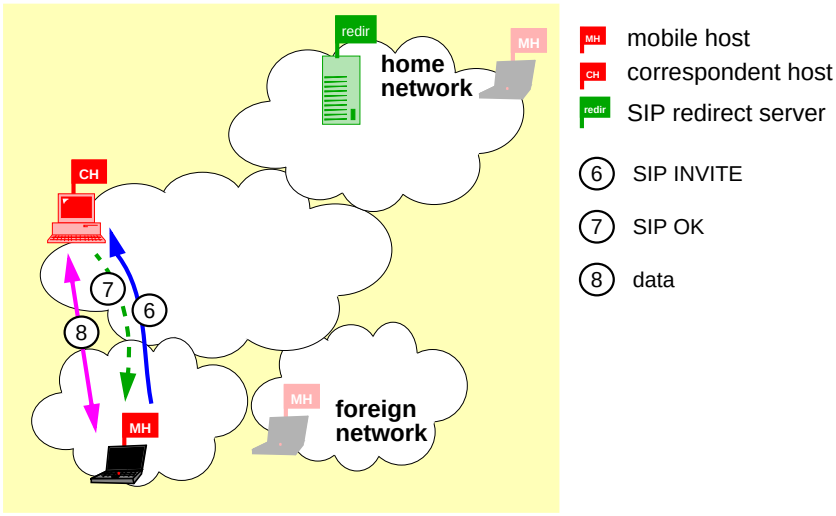

Figure 6. SIP Mobility: Mobile Host Moves

The SIP INVITE (step 6 in Fig. 6) request could look as follows:

```
INVITE sip:alice@correspondent.com SIP/2.0
Via: SIP/2.0/UDP mh.current.location:5060
From: sip:betty@home.com
To: sip:alice@correspondent.com
Subject: a mobile session
Contact: betty@mh.current.location
CSeq: 781769870 INVITE
```

```
Call-ID: <call-id of ongoing session>
Content-Length: 192

v=0
o=betty916340046 916340046 IN IP4 mh.current.location
s=No Title
i=No Information
t=2208988800 2208988800
c=IN IP4 mh.current.location
m=audio 50000 UDP 0
```

Betty owns the mobile host, and Alice is the user at the correspondent host. Betty's regular address (betty@home.com) is used in the "From" field, since that is used for identification, and can also be used as a fall back mechanism in case the communication is lost (more discussion on this in section 3.3). The new address (mh.current.location) is put in the "Contact" field, and in the "c=" field of the session description (SDP) part of the message. For more information on the different fields, please consult [4] and [8].

3.2 SIP Mobility Support with Mobile IP

If the mobile host is using mobile IP, it is not necessary, albeit useful, for the SIP server to have knowledge about the current location of the mobile host. One solution to avoid duplicate information is to co-locate the SIP redirect or proxy server and the home agent, or to allow the SIP server to query the home agent about the location of the mobile host. It would also be possible to actually send the invitation to the home address, let the home agent forward the invitation to the correct location, and let the mobile host provide information about its location in the response, using the "contact" header.

3.3 Error Recovery

If the correspondent host for some reason has an outdated address of the mobile host, it must have a fall-back mechanism to break the error situation. One example of this is when we have two mobile hosts having a conversation, both loose contact for a while (e.g., enter a tunnel), and when they gain contact again, they have both got a new IP address. See Fig. 7.

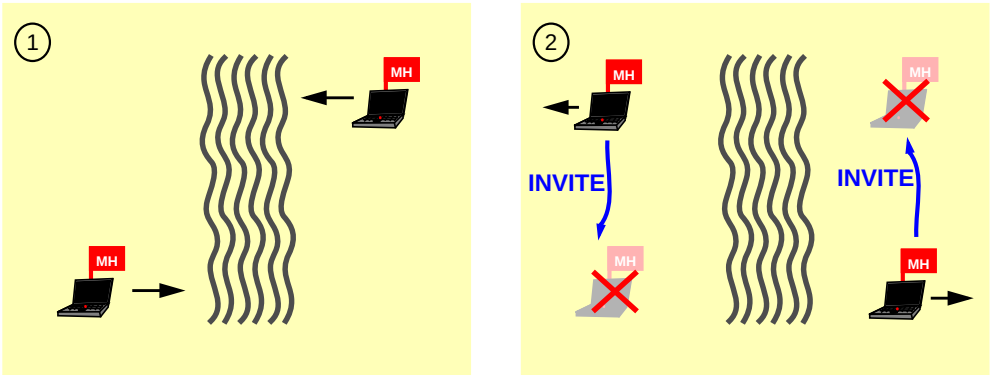

Figure 7. Stale Address Situation

In order to avoid situations like this, a host can send retransmissions of invitations to a mobile host to the SIP server on the mobile host's home network. Since the SIP server has a fixed address, the mobile host can always make registrations to it. In that way, the correspondent host can re-locate a mobile host that has been lost.

3.4 Security

In the SIP specification there is support for both authentication and encryption of SIP messages, using either challenge-response or private/public key cryptography.

3.5 RTP Mobility Support

It is possible to further enhance the mobility support by using mechanisms in the Realtime Transport Protocol (RTP). RTP is used for transporting realtime traffic over IP, and is always used in conjunction with the Realtime Control Protocol (RTCP), which is used for quality feedback and exchange of user information. RTCP packets are sent periodically within a media stream. Associated with an RTP media stream are:

- SSRC, Synchronization source, which is a 32 bit integer identifying the source of the RTP stream. It is unique within one RTP session, and is carried in every RTP packet. Different media from the same source use different SSRC numbers.
- CNAME, which is a transport level identifier used in RTCP to identify the user, since a user may have several RTP streams. It has the format "user@host", where host is a fully qualified domain name or IP address. It is carried in an RTCP packet together with the corresponding SSRC identifier.

In packets containing mixed data, the SSRCs of the contributing sources are put in the list of Contributing Sources (CSRC) in the RTP header.

The SSRC is picked randomly by each host, and the probability of the SSRC being unique within a session is very high. In addition, there is a collision and loop detection mechanism, which is used to ensure that the SSRC of each stream is unique within the session, and that there are no loops introduced, e.g., by media mixers sending a stream back to a source. The collision detection mechanism that each host performs states that a collision has taken place if the source finds that two RTCP packets with source descriptions have the same SSRC but different IP addresses and different CNAMEs. For the loop detection, every host keeps a list of conflicting addresses, which are addresses of sources that have been sending RTP packets using the same SSRC or CSRC as the host itself. The host considers the packets coming from addresses in the table as looped, and disregards them.

Thus, each RTP and RTCP packet contains the sender address and the SSRC identifier. If an RTP sender changes IP address while it is sending packets, the receiver will see packets with a new IP address, but the same SSRC. The correspondent host realises that the mobile host has a new address, and redirects its packets to that address, see Fig. 8.

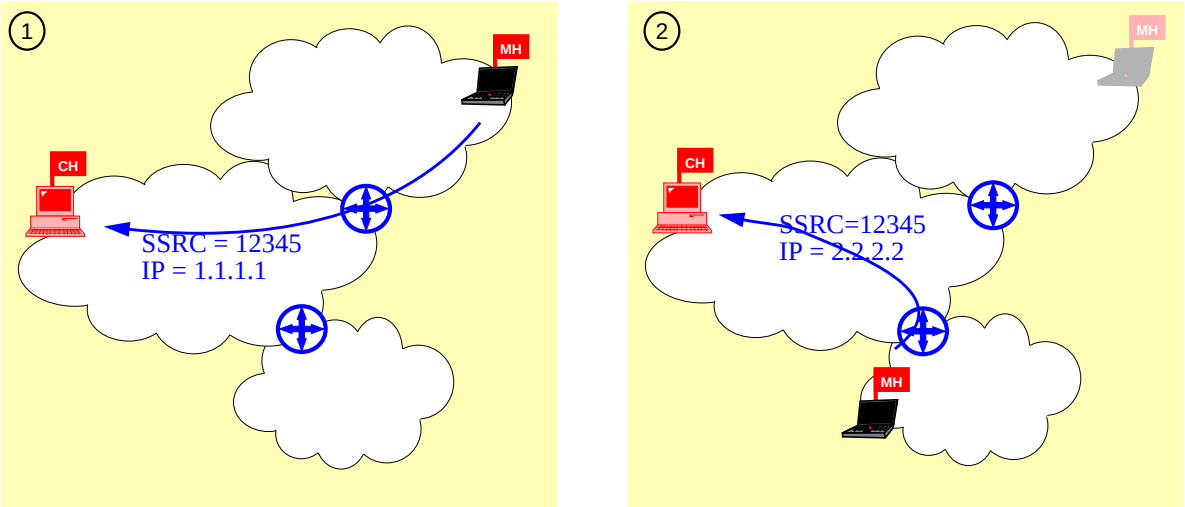

Figure 8. RTP Mobility

There are two problems with this simple approach: One is the collision and loop detection mechanism. However the collision detection mechanism will not consider the above scenario a collision or loop because it uses the RTCP packets, and since the CNAME does not change, the change of IP address will not be considered a collision. The second issue is security, since a malicious user could steal a flow by starting to send with the same SSRC as an existing member of a session. However, if the malicious user can see the stream, there is no reason to steal it, except for denial of service attacks. If the malicious user cannot see the flow, he must guess a 32-bit SSRC, which is not trivial. If a flow needs high protection, encryption should be used to protect the stream.

It is worth noting that this kind of mobility support is only useful when both the mobile host and correspondent host are sending RTP packets.

4 Dynamic Host Configuration Protocol

The Dynamic Host Configuration Protocol (DHCP) [10] is a client-server protocol used to dynamically assign IP addresses and providing network information to internet hosts. For mobile hosts, DHCP can be used to assign a co-located care-of address, in the case when no foreign agent is used.

The basic message exchange needed for acquiring an IP address is shown in Fig. 9.

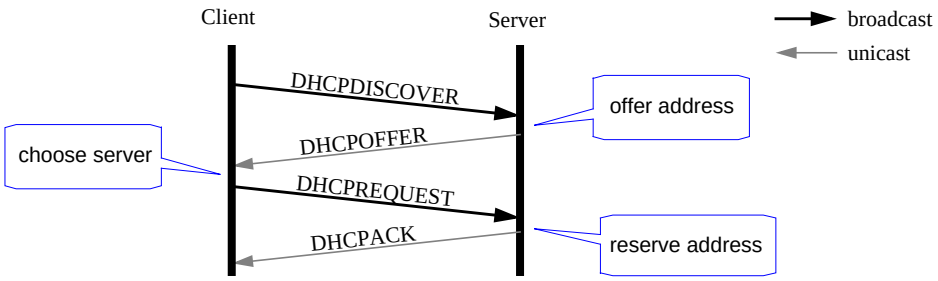

Figure 9. DHCP Message Exchange

The client broadcasts a DHCPDISCOVER message, in order to discover a DHCP server. Upon reception of a DHCPDISCOVER message, a DHCP server checks if it has an address to offer the client. Several DHCP servers may answer by sending a DHCPOFFER with the address they are offering, together with other network information to the client. The client chooses one of the offers and broadcasts a DHCPREQUEST message, containing the chosen IP address. The DHCP server can then see whether the client is requesting the address the server was offering. The address is definitely assigned to the client when the chosen server sends a DHCPACK message. In most cases there will only be one DHCP server answering the DHCPDISCOVER.

The DHCP client is allowed to use the IP address it has received from the DHCP server for a limited amount of time, specified in the DHCPOFFER and DHCPACK messages. If the client wants to use the address after this time, it must renew the lease. If a client does not need the address even though there is time left on the lease, it may release the IP address by sending a DHCPRELEASE message to the server. Currently, there is no security mechanism in DHCP, but there is ongoing work on this issue within the IETF [11,12,13].

For mobile hosts, it is important to get an IP address quickly after entering a new network. In the DCHP specification, it says that a DHCP client should wait for one to ten seconds before sending the DHCPDISCOVER message. This is to avoid flooding the DHCP server with messages, e.g., if there has been a power-down, causing all hosts on the network to start up simultaneously. For mobile hosts, this is not a significant problem since the host would send the DHCPDISCOVER when it enters a new network. Because of this, and because we want to minimize the handoff delay, a mobile host should send the DHCPDISCOVER as soon as it knows it is in a new network.

When a mobile host leaves a network, it should send a DHCPRELEASE so that the IP address is not reserved for a period longer than necessary. However, a mobile host does not usually know that it is leaving a network until it is too late. Either the mobile host must be able to send a DHCPRELEASE after it has left the network, or the lease time should be tuned to release the addresses sufficiently fast when they are no longer needed. The lease time should not be too short either, because it may cause a large amount of DHCP messages sent over the network for updating a lease.

5 Proposed Architecture

As described earlier, there are several problems with using mobile IP for real-time traffic, since we can expect the mobility function to add jitter and delay to the data streams. The jitter is introduced during handoff, when the mobile host registers with a new foreign agent and sends a registration to the home agent. By using RTP mobility, the handoff can be made faster, minimizing the introduced jitter.

It is stated in [1] that mobile IP is intended for slow mobility, or macro mobility[1], but even with this condition, there are problems with both delay and bandwidth efficiency. The encapsulation adds at least 12 bytes to each packet, which is a significant amount for speech packets, which are usually small. The delay introduced when packets are routed via the home agent also affects delay sensitive realtime traffic in particular.

1.Host is moving between IP subnets, not just changing base stations covering a small area.

By introducing SIP mobility support, these problems are solved to a large extent. However, in order to support both TCP connections and realtime communications, we would like to use both mobile IP *and* SIP mobility support. This can be achieved if we allow the mobile host to choose when to use its home address or care-of address. When sending RTP streams it will use the care-of address, and when establishing TCP connections, it will use the home address and let the traffic go via the home agent. It may also use route optimization for the TCP connections.

What we propose is a similar solution to the one presented in [9], which is to use mobile IP when necessary for long-lived TCP connections such as telnet, ftp, etc. For other connections, regular IP is used. For services such as web browsing, the connections are usually short enough to give a small probability that a handoff breaks a connection, and if that would happen, the user can push the reload button on the browser to resume activity. This is also helped by the fact that users have low expectations on the web service. For realtime connections, the mobility is supported by RTP and SIP. The protocol stack is described in Fig. 10. As in [9], a mobile policy table is used for deciding what source address to use (home or care-of address), whether it should be tunnelled, or even use a bidirectional tunnel.

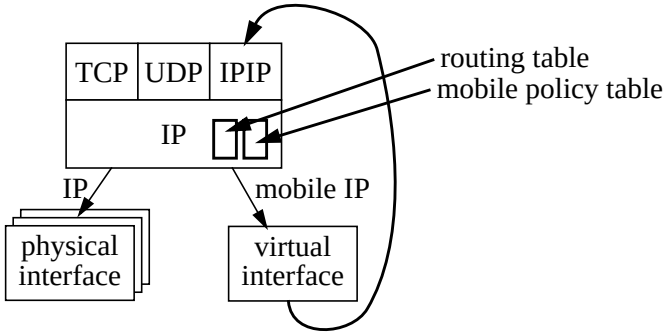

Figure 10. Protocol Stack

6 Performance

It is not trivial to compare the performance of mobile IP vs. SIP mobility, because it very much depends on the distance between the mobile host, correspondent host, and the mobile host's home network.

6.1 End to End Delay

It is obvious that the delay will be lower if packets are sent directly to the mobile host without being routed via the home network and/or being encapsulated. The extra latency is basically proportional to the distance to the home network and the correspondent host. The delay introduced by the home agent and foreign agent are relatively small unless a congestion occurs and packets are buffered.

6.2 Handoff Delay

The handoff delay depends on several different parameters:

- The time before the mobile host realizes it is in a new network, t_{real},
- the time it takes to get an IP address (only when a foreign agent is not used), t_{addr},
- for mobile IP, the time it takes to register with foreign agent and/or home agent, t_{faha}, and

- for SIP mobility, the time it takes to send an INVITE to the correspondent host (only when SIP mobility is used), t_{invite}.

In addition, for SIP mobility, the mobile host must register with the SIP server on the home network, even if this is not part of the handoff delay.

For mobile IP when a co-located IP address is used, the handoff delay is

$$t_{mobIP} = t_{real} + t_{addr} + t_{faha} ,$$

whereas for SIP mobility, the handoff delay will be

$$t_{SIPmob} = t_{real} + t_{addr} + t_{invite} .$$

As can be seen, the difference in delay is

$$t_{SIPmob} - t_{mobIP} = t_{invite} - t_{faha} .$$

The difference depends on how fast the implementations are, and the distance to the home network and correspondent host are, respectively. What this shows is that SIP mobility does not necessarily increase the handoff delay, at the same time as it always reduces the latency for the data traffic compared to mobile IP. For the RTP mobility, the gain is that the update of the address is indirect and automatic, without the need to create and parse SIP messages, which can make the handoff faster.

7 Related Work

There is much work being done regarding IP mobility support. The work that has had the most impact on this paper is the Mosquitonet project at Stanford [9]. In this project they are providing the Mobile Policy Table (MTP) which is used to define when mobile IP is to be used. However, other mobility support than mobile IP is not used in this project.

In [14], cellular IP is proposed to be used for mobility within IP subnets, in order to do faster handoffs. This could be used together with the SIP mobility and/or mobile IP. Another approach to faster handoffs is [15], which proposes regional aware foreign agents.

8 Future Work

8.1 Implementation

The SIP mobility support is currently being implemented. The mobile host has a mobile host daemon, which is the one implemented in the Mosquitonet project, and a SIP client based on tcl/tk. Moreover, the host is equipped with a wavelan interface, and is using DHCP for getting an IP address.

It would be desirable to also include a lower level support, e.g., the cellular IP implementation [14] together with the SIP mobility and mobile IP in order to have a complete solution.

8.2 Hierarchical Servers

When the mobile host is far away from its home network, sending a new registration to the home SIP server every time it moves can place an unnecessarily high load on the SIP server and network, especially if the home SIP server is serving many hosts. Instead, the mobile host can register with a closer SIP server, and the SIP server on the home network knows to which SIP server it should forward/redirect an incoming request. The basic approach is similar to both [14] and [15], but in this case, only the first SIP messages need to be routed via the servers in the hierarchical path. Future SIP messages and the media stream can then be sent directly between the hosts.

9 Conclusions

This paper has proposed the use of mobility support in SIP and RTP for realtime communication. The intention is not to fully replace mobile IP, but to use each scheme when applicable.

Using SIP for mobility is possible without making any changes to the IP stack of the mobile host. If we want to support mobile IP as well, we use a mobile policy table for deciding when to use the home or care-of address in addition to the changes needed to support mobile IP itself. It is important to point out that unless route optimization should be supported, no changes to the kernel are needed for the correspondent host. The RTP mobility support may be used to enhance the handoff performance, although it is only useful when both the mobile and corresponding host are sending packets.

What applies to both mobile IP and SIP/RTP mobility is that none of them is suitable for fast, or small scale mobility. The fast mobility should either be supported by lower layers or by some other, more suitable, scheme. One suggestion is Cellular IP [14], which can be used together with mobile IP or the SIP scheme.

10 References

[1] C. Perkins, “IP mobility support,” Internet Request for Comments 2002, Internet Engineering Task Force, Oct. 1996.

[2] C. Perkins, “Route Optimization in Mobile IP,” Internet draft, Internet Engineering Task Force, Nov. 1997. Work in Progress

[3] H. Schulzrinne, S. Casner, R. Frederick., and V. Jacobson, “RTP: A Transport Protocol for Real-Time Applications,” Internet Request for Comments 1889, Internet Engineering Task Force, Jan. 1996.

[4] M. Handley, H. Schulzrinne, E. Schooler, and J. Rosenberg, “SIP: Session Initiation Protocol,” Internet Draft, Internet Engineering Task Force, Aug. 1998. Work in Progress.

[5] R. Pandya, “Emerging mobile and personal communication systems,” IEEE Communications Magazine, vol. 33, pp. 44-52, June 1995.

[6] C. Perkins, “IP Encapsulation within IP,” Internet Request for Comments 2003, Internet Engineering Task Force, Oct. 1996.

[7] C. Perkins, “Minimal Encapsulation within IP,” Internet Request for Comments 2004, Internet Engineering Task Force, Oct. 1996.

[8] M. Handley, and V. Jacobson, “SDP: Session Description Protocol,” Internet Request for Comments 2327, Internet Engineering Task Force, Apr. 1998.

[9] X. Zhao, and M. Baker, “Flexible Network Support for Mobility,” ACM/IEEE Mobicom’98, Oct. 1998.

[10] R. Droms, “Dynamic Host Configuration Protocol,” IETF Request for Comments 2131, Internet Engineering Task Force, March 1997.

[11] R. Droms, W. Arbaugh, “Authentication for DHCP Messages,” Internet draft, Internet Engineering Task Force, Aug. 1998. Work in Progress.

[12] O. Gudmundsson, R. Droms, “Security Requirements for the DHCP protocol,” Internet draft, Internet Engineering Task Force, March 1998. Work in Progress.

[13] S. Drach, “DHCP Option for User Authentication Protocol,” Internet draft, Internet Engineering Task Force, Sept. 1998. Work in Progress.

[14] A. G. Valko, “Cellular IP - A New Approach to Internet Host Mobility”, to appear in ACM Computer Communication Review, January 1999.

[15] S. F. Foo, and K. C. Chua, “Regional Aware Foreign Agent (RAFA) for Fast Local Handoffs”, Internet draft, Internet Engineering Task Force, November 1998. Work in Progress.