SIP and MIPv6: Cross-Layer Mobility

Rui Prior, *IEEE Student Member*
IT — University of Porto
rprior@dcc.fc.up.pt

Susana Sargento, *IEEE Member*
IT — University of Aveiro
ssargento@det.ua.pt

Abstract

Terminal mobility may be handled at different layers. Though the MIPv6 protocol is the strongest candidate for handling mobility in next generation networks, mobility management facilities are also provided by SIP, the most widely deployed and researched protocol for session control. When jointly used, this duplication of functions leads to inefficiencies in session setup signaling, particularly if coupled with end-to-end resource reservation for the media flows. This paper analyses these inefficiencies and proposes an integrated approach that minimizes the session setup delay. The gains of the proposal are demonstrated both by a delay analysis and by simulation results.

1. Introduction

Current research, standardization and market trends indicate that future telecommunication systems will be based on IP, representing a convergence of actual networks, services and applications onto a single infrastructure. This convergence requires the integrated support of different access technologies, mostly wireless. Moreover, users must be allowed to move freely without disruption of their ongoing sessions, even when the movement leads to a change in the access technology. While the use of IPv6 with mobility support [3] as a convergence layer greatly simplifies this process, service provisioning with the necessary quality and seamless mobility in such heterogeneous scenario is still a heavily researched topic.

The protocol that will most likely be used for the initiation and control of multimedia sessions is SIP (Session Initiation Protocol) [11][4], which has been adopted by the principal 3G standardization organizations and forums, like 3GPP and 3GPP2. Though SIP itself may be used for mobility management [14], this function is better handled at layer 3 by Mobile IPv6 (MIPv6) [3] (even when SIP is used for session control) for several reasons: (1) applications need not worry about mid-session mobility unless serious changes in available resources force a session renegotiation, e.g., to a lower bitrate codec; (2) layer 3 mobility support must be in place to support non-SIP sessions (HTTP, FTP, etc.), and uniform mobility management is desirable for robustness and flexibility; and (3) seamless mobility may be achieved udingMIPv6 extensions like Fast Handovers (FHO) [5].

The joint use of SIP and MIPv6, however, leads to some inefficiency issues in pre-session mobility, due to each protocol's unawareness of the other's mobility management capabilities. The issues are even worse when end-to-end resource reservation must be performed to ensure appropriate Quality of Service (QoS) to the session, requiring knowledge of the points of attachment of both terminals. This inefficiency may lead to a significant delay in session setup, especially in the presence of packet loss (not uncommon in wireless links) and of large round-trip times (RTT). This paper proposes a scheme for the minimization of these delays based on simple procedures and cross-layer interactions, making SIP aware of the terminal's location, that is, the Care-of Address (CoA).

The paper is organized as follows. Next section describes some previous work on the integration of SIP and MIP. Section 3 gives an overview of the target architecture for these optimizations. Sections 4 and 5 contain an analysis of the problem and the proposal of the solution, respectively. Section 6 describes the SIP registration procedures. An analytical comparison of the standard and optimized procedures is presented in section 7, and section 8 discusses simulation results of both. Finally, section 9 contains the main conclusions.

2. Related Work

Different degrees of integration of SIP and MIP (v4 or v6) have been proposed by several authors. Jung et al. [4] proposed the use of integrated mobility agents for SIP and MIP(v4). Some MIP functions (like binding refreshments) are transposed to SIP, and mobility is communicated to the correspondent nodes (CNs) via re-INVITE requests. This approach imposes different handover procedures for SIP and non-SIP sessions

(UDP or TCP), and the security issues of establishing bindings with CNs via SIP were not addressed.

Politis et al. [8] proposed a hybrid SIP/MIP(v4) scheme for inter-domain mobility. Their approach avoids the IP-in-IP MIP encapsulation for SIP sessions, but not for non-SIP ones. Their work mostly concerns mid-session mobility which, in our case, is handled by a modified MIPv6 with FHO. Moreover, the encapsulation problem is mitigated in MIPv6 by the use of routing optimization.

Wang et al. [12][13] proposed an integrated SIP-MIP mobility management architecture, where MIP and SIP agents are broken down into functional blocks and then integrated without duplication into unified Home and Foreign Mobility Servers (HMS/FMS). Their proposal mostly intends to solve the problems associated with different types of mid-session mobility. Moreover, their architecture is different from ours in that it requires one FMS per (access) network.

None of these proposals addresses the issues with the integration of end-to-end resource reservation with session signaling.

3. Architecture overview

This section contains a brief overview of the network architecture targeted by these optimizations. The routing infrastructure is based on IPv6, and mobility is supported at layer 3 by MIPv6 with FHO extensions.

The network is divided into administrative domains, each consisting on a number of access networks (ANs), possibly with different access technologies, interconnected by a core network. One of the key components is the QoS Broker at the AN, responsible for controlling the admission of flows and the handovers. The QoS Brokers have information on available resources not only for the AN they control, but also in the core of their domain and the transmission direction of the inter-domain path segment. Therefore, the combined admission control performed at the caller and callee sides may ensure that enough resources are available along the end-to-end path (fig. 1). In the core network there is a Multimedia Service Platform (MMSP), consisting of a broker and SIP proxies, responsible for controlling multimedia services. Please refer to [9] for more information on the QoS subsystem of the architecture.

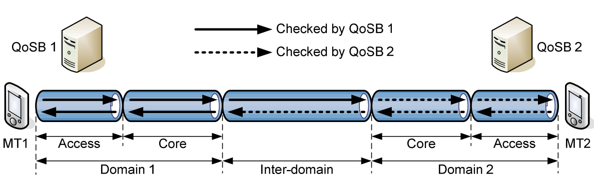

Figure 1. Admission control (inter-domain call)

Since energy is a scarce resource in mobile terminals, the system supports a dormancy mode for energy saving. An alternate CoA is provided to the Mobile Terminal (MT) by a Paging Controller (PC) before it enters dormant mode; when packets arrive, the PC buffers them and informs the MT; when the MT wakes up, the buffered packets are delivered and the MT starts using its new, real CoA (more details in [1]).

In order to establish a reservation for a flow with end-to-end QoS, admission control needs to take into account the available resources in the complete path, including the access, core and inter-domain path segments. To this end, each mobile terminal must be aware of its correspondent's physical location which, in IP terms, corresponds to its CoA. SIP's unawareness of pre-session MIPv6 mobility, as will be seen in the next section, is one of the sources of inefficiency in session initiation signaling. Mid-session mobility, on the other hand, is handled by MIPv6 with FHO, and does not require intervention of SIP.

4. Inefficiency of SIP with MIPv6

In this section we analyze the inefficiencies of the joint use of SIP and MIPv6, particularly in an environment where end-to-end resource reservations must be performed. The message sequence for initiating a call between two roaming terminals is illustrated in fig. 2. "100 Trying" SIP responses and "PRACK" requests and responses have been omitted in the figure, since they are not in the critical path of signaling.

The sequence is initiated by the caller sending an INVITE with a message body containing an offer with the set of codecs supported by the caller and the corresponding ports (at the caller end only); this message is sent via the outbound proxy, MMSP1.f. If the binding cache of MMSP1.f is not up to date with the caller's current CoA, this INVITE is tunneled to the Home Agent (HA1), from where it is sent to MMSP1.f, introducing an additional delay corresponding to one round trip time (RTT) between the caller's home and foreign domains. The caller may then initiate a return routability procedure (RRP – grayed out since it is not in the critical path of signaling) to MMSP1.f so that further messages between them are optimally routed. If the Home Keygen Token has not expired since registration, only the Care-of Test Init/Care-of Test exchange is necessary; otherwise, a full RRP must be performed. Notice that mobility-unaware applications use Home Addresses (HoA) as endpoints in order for layer-3 mobility to be transparent.

When the INVITE request arrives at MMSP1.f, it must find out the proxy responsible for the callee to send the INVITE. To this end, a DNS lookup is per-

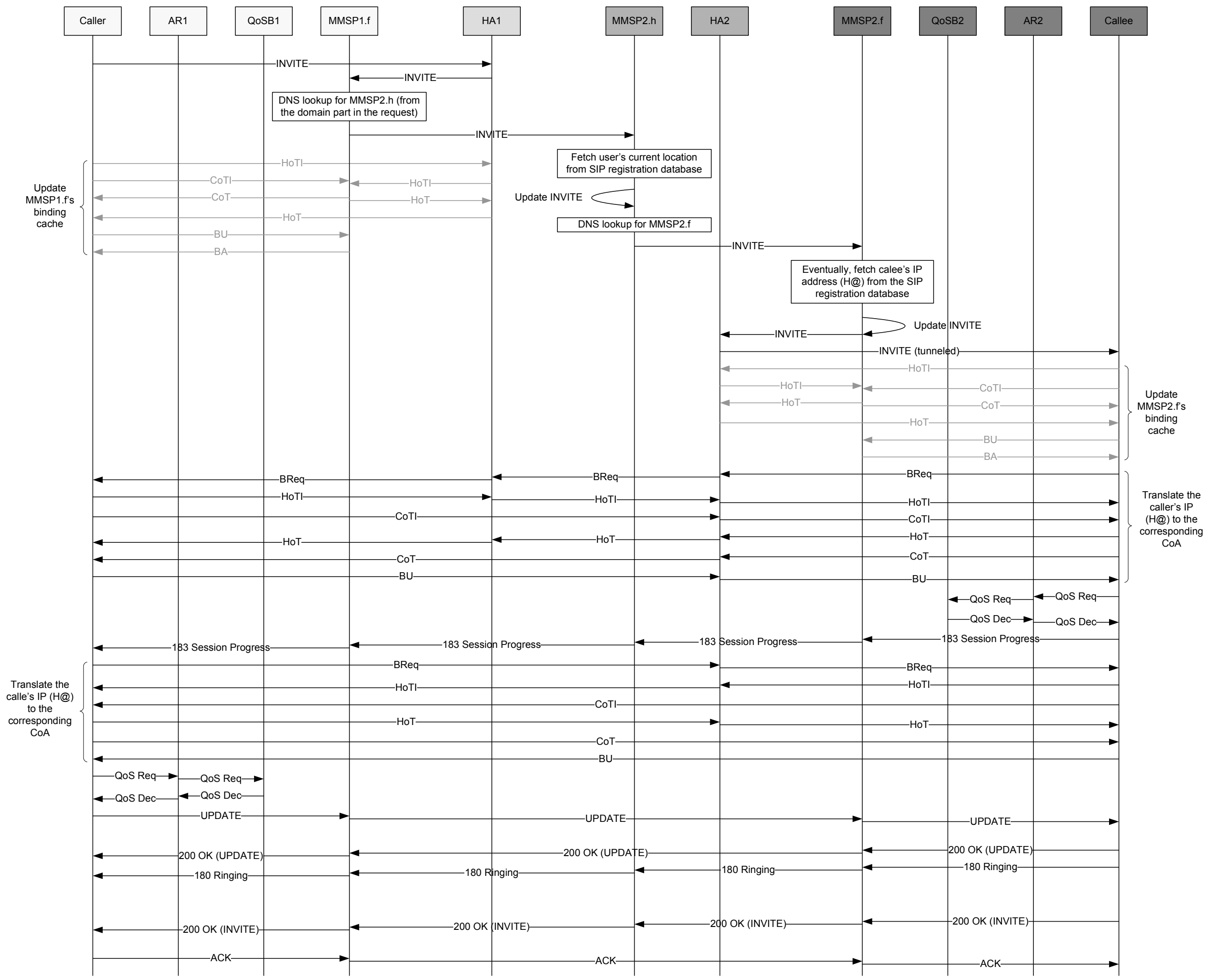

Figure 2. Inter-domain call without optimization (both terminals roaming)

formed, involving a round-trip to a root DNS server, another one to a top level DNS server, and one or two[1] to the home domain of the callee, unless the entries are already cached. On receiving the INVITE, MMSP2.h looks up the registration database and finds out that the user (callee) is roaming; DNS lookups are performed to find out the proxy for the foreign (visited) domain. Notice that service authorization is mandatory, therefore MMSP2.h cannot send the INVITE directly to the callee – packet filtering mechanisms would drop it.

MMSP2.f receives the INVITE and fetches the callee's IP address from its registration database (for the sake of simplicity, we assume that the callee has registered itself with the IP address rather than a host-name). Since regular SIP is not layer-3-mobility-aware, this IP address is a HoA; therefore, the message must go to the callee's HA, where it is tunneled to the callee.

[1] At least an SRV lookup, but usually preceded by a NAPTR lookup.

When the callee receives the INVITE, it builds a list of the common codecs. In possession of the IPs and ports at both ends (the caller and itself), it may request resources to/from the caller (QoS Req). However, if resources are reserved for more than the wireless link, as in our case, the reservation must be made according to the physical points of attachment of the terminals, that is, their CoAs. The callee knows its own CoA, but not the caller's. Therefore, a Binding Request (BReq) must be issued to the caller, which will trigger a return routability procedure and a Binding Update (BU) from the caller to the callee, adding two RTTs between them. Since all these messages go through the HA of the callee (the caller has no binding for the callee yet) and some of them (BReq, HoTI and HoT) through the HA of the caller, this translates in 11 inter-domain traversals (considering that HoTI/HoT, not CoTI/CoT, are in the critical path of signaling, as is most common).

The callee also initiates a return routability procedure and binding update to MMSP2.f, so that future messages need not be tunneled; however, the 183 Session Progress response must still go through the HA (otherwise MMSP2.f would drop it, since it has no binding for the callee).

When the caller receives the 183 Session Progress, it knows its own CoA, but not the callee's; therefore it must send a BReq to the callee (symmetrical of the previously mentioned procedure). Two additional RTTs are, therefore, added (corresponding to 7 inter-domain traversals, not 11 as the previous one, since the callee already has a binding for the caller). Only now the HoA and CoA of the callee are known at the caller side, therefore only now a fully-formed QoS request may be performed at this side. As the amount of available resources may be less than what was reserved at the callee side, an indication of the final codec configuration (counter-answer) must be sent in an UPDATE request in order to synchronize the reservations. Hopefully, by this time all the binding caches are updated, meaning that the UPDATE (as well as all further signaling) travels through optimal paths. The media packets will also use the optimized path, since each terminal has a binding for the media address of the other one (which is usually the same as the signaling address, except in some multi-homed terminals).

Many of the inefficiencies are due to the SIP protocol's unawareness of layer-3 mobility, and to the need to perform resource reservations combined with this unawareness. As we will see in the next section, this scenario can be much improved by means of very simple procedures and cross-layer interactions.

5. Optimizing the use of SIP with MIPv6

The first optimization consists on eliminating the need for the INVITE message between the caller and MMSP1.f to go through the HA. While this could be easily accomplished by having the terminal keep the MMSP's binding cache updated all the time, such approach would lead to a lot of unnecessary signaling, since most of the time it is not actually communicating, and would limit its ability to conserve energy using the dormancy/paging features of the system. Therefore, we propose a different approach: using the CoA as source IP address of the packet containing the INVITE message. Notice that the INVITE message itself still uses the HoA. Responses to the INVITE will be delivered to the CoA, since the proxy adds a *received* parameter with the source IP address of the packet to the *Via* header of a received request, whenever the *sent by* parameter in the *Via* header does not match that source IP address (in our case it contains the HoA). The terminal may then perform the RRP, which is not on the critical path of signaling, and then maintain MMSP1.f's binding cache updated for the whole duration of the call, so that future requests (PRACK, UPDATE, ACK, re-INVITEs, etc.) and their respective responses will always use the optimized path.

The goal of the second optimization is to eliminate the need for the INVITE message between MMSP2.f and the callee to go through the callee's HA. Contrary to the previous case, the message is not generated at the mobile terminal. In order to use the callee's CoA as the destination address, MMSP2.f must have knowledge of the mapping between the callee's HoA and CoA, as the HoA is the one used by the application layer. In order to provide this information, we introduce a cross-layer interaction at MMSP2.h: after retrieving the IP address (HoA) of the callee from the registration database, MMSP2.h queries the HA to find out the callee's current CoA. The URI in the request line is then changed to the IP address (HoA), as usual, but with tag containing the current CoA (e.g., "coa=FF1E:03AF::1") appended. Using the CoA from the tag in the request line as the destination IP address of the packet, MMSP2.f may send the INVITE directly to the callee. Notice that the use of the CoA tag by MMSP2.f for direct forwarding does not add any security issue to standard SIP, since the same would occur if MMSP2.h had placed the CoA directly in the request line of the forwarded INVITE.

The third optimization concerns the elimination of the DNS lookup at MMSP2.h when forwarding the INVITE request: if the registration for redirection includes the IP address of the inbound proxy where to forward an incoming INVITE (MMSP.f, in this case), no DNS lookup to find this proxy is necessary. The use of the Path header field described in [15] is recommended for conveying this information, while also providing a simple means of enforcing the traversal of an MMSP at the foreign domain, necessary to perform service authorization and filtering.

The fourth optimization is related to the need to perform network resource reservations concerning more than the wireless link. Since the requests are performed for a path-optimized flow, they must be performed between the physical locations (CoA) of both terminals. Once again, we rely on the transport of CoA information in application signaling. However, since there is no guarantee that the media will use the same IP addresses as SIP signaling (particularly with multi-homed terminals), the CoA information used to this end is conveyed not in SIP, but in the protocol used for session negotiation (Session Description Protocol - SDP [2] or SDPng [6]).

One might argue that the inclusion of layer 3 mobility information in an application protocol such as SIP should not be done because it breaks the layering

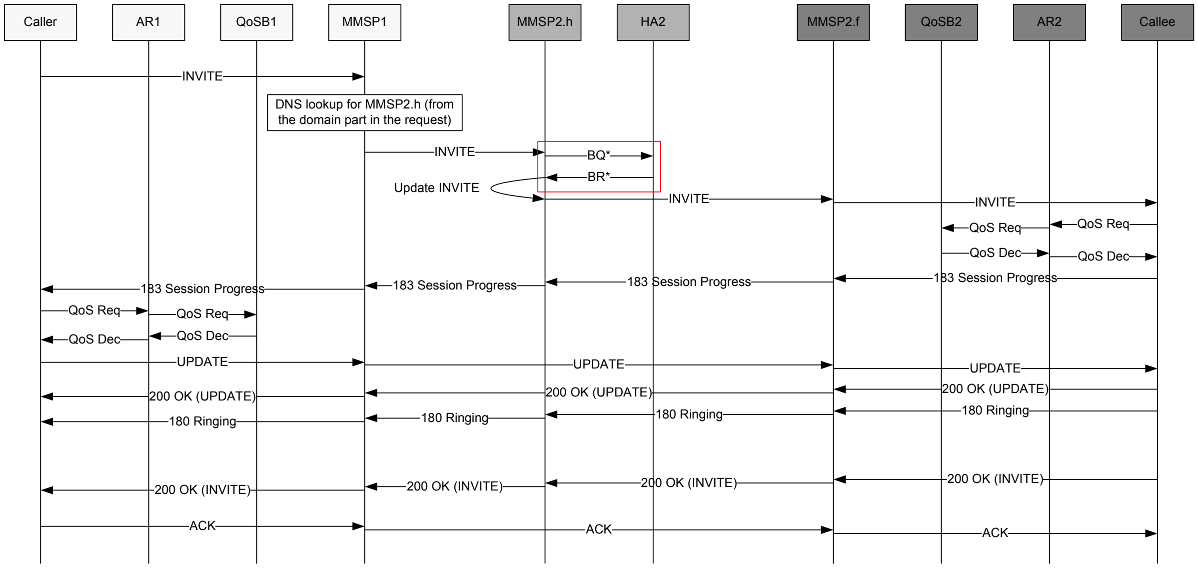

Figure 3. Optimized inter-domain call (both terminals roaming)

principle; it is worth noting, however, that not only does the standard SIP already include layer-3 information (IP addresses) in its headers, but also that cross-layer information would be required by any protocol with similar characteristics, namely regarding independence between the signaling and media interfaces.

5.1 Optimized initiation sequence

Figure 3 shows the message sequence for an optimized call, with both terminals roaming (messages not in the critical path of signaling are omitted). Since the INVITE is sent with the CoA as source IP address, it goes directly to MMSP1.f. A DNS lookup is performed (there is no way to avoid it) and the message is forwarded to the callee's home proxy. MMSP2.h changes the request line from the URI to the HoA of the callee, adding a *coa* tag with the callee's current CoA (retrieved from HA2) to the request line of the INVITE. Although in the optimal case MMSP2.h and HA2 would be integrated, with the respective location databases merged, even if they are not, communication between them is fast and efficient (they belong to the same domain and are located close to one another). This communication, however, requires new messages, Binding Query (BQ) and Binding Response (BR), since the standard Binding Refresh Request (BRR) and BU messages are exchanged with the MT, not the HA.

Since MMSP2.h has the IP address of the callee's outbound/inbound proxy, MMSP2.f, there is no need for a DNS lookup. Using the information from the *coa* tag on the request line, MMSP2.f is able to send the INVITE directly to the callee without the need to go through its HA. When this message arrives, the callee retrieves the caller's HoA and CoA from the SDP, and uses this information to request network resources.

After receiving the reservation response, the callee sends a 183 Session Progress response, containing an answer with the set of common codecs and their respective ports at both ends, to the caller. Information on the callee's CoA is included in the SDP; this information is used by the caller to perform the resource reservation on its side.

Usually, by the time the UPDATE is sent, both terminals have already established bindings with their respective proxies. However, the caller may include a *coa* tag with the CoA of the callee to the request line, lest MMSP2.f not have yet a binding for MT2: if this is the case, MMSP2.f uses the tag to send the request directly to the CoA, as it has previously done with the INVITE; otherwise, the tag is ignored. Notice that the UPDATE (and all further requests) does not traverse the home proxy of the callee, since only the local (foreign) proxies, with responsibilities in service control, have added themselves to the *Record-Route* header of the INVITE.

It is worth noting that bindings must still be established between the terminals for the media sessions, meaning that the overhead of both solutions will be comparable (except for a few encapsulated packets in the standard signaling); however, these message exchanges are moved out of the critical path of signaling in the optimized case.

Mid-session mobility is handled exclusively at layer 3 by MIPv6 with FHO. SIP sessions are handled similarly to non-SIP ones, and no re-INVITE message is sent unless a session renegotiation (e.g., for changing the codec or bit rate) is necessary.

6. SIP registration

A user at home registers normally with the local (home) proxy, using the Address of Record (AoR) in the *To* header and the IP address (HoA) in the *Contact* header. A roaming user must register itself with the foreign MMSP, since it will be performing service control, but also with the MMSP of its home domain for location purposes. In the standard case, the user registers itself with MMSP2.f as user@home.com, using the IP address as *Contact*; MMSP2.f changes the *Contact* to user%40home.com@foreign.com and forwards the registration request to MMSP1.h. In the optimized case, the user registers itself with MMSP2.f as user@home.com using the IP address as *Contact*, similarly to the standard case, but instead of changing the *Contact*:, MMSP2.f adds a Path header [15] with its own IP address, forcing incoming requests from MMSP1.h to traverse it. Though the Path header is an extension to the basic SIP protocol, it is a standard one.

7. Delay analysis

In this section we perform a comparative analysis of the dial-to-ringtone delay with standard and optimized signaling for a call between two roaming terminals. The following assumptions have been made:

- Inter-domain delays are symmetrical – it takes about the same time to go from A to B as from B to A.
- Compared to the delay in the wireless link or in inter-domain trips, the delay in intra-domain wired links is minimal and may, therefore, be neglected.
- In the RRP, the HoTI/HoT exchange takes longer than the CoTI/CoT.
- An RRP from a terminal to a local MMSP takes less time than the same procedure to a remote terminal, provided the HA is the same in both cases.
- The BU from the Caller arrives at MMSP1.f before the 183 Session Progress in the sequence of fig. 2, meaning that the 183 Session Progress will not go through the HA even in the standard case.

We will use the following notation: T_{W1} and T_{W2} are the delay at the wireless links of the caller (1) and the callee(2); T_{F1F2}, T_{F1H1}, T_{F1H2}, T_{F2H1} and T_{F2H2} are the inter-domain one-way trip delays (between combinations of the Foreign and Home domains of the caller and the callee); T_{DNS1} and T_{DNS2} are the delays for DNS lookups of the home and the foreign domains of the callee, respectively. Notice that if the entries are not cached, the DNS lookups imply at least one RTT to the DNS registrar, to find out the DNS server of the domain to be resolved, and another one or two to that domain, to find out the address of a SIP proxy.

7.1 Standard Case

With standard, non-optimized signaling (refer to fig. 2), the INVITE takes

$$T_{Inv} = T_{W1} + 2T_{F1H1} + T_{DNS1} + T_{F1H2} + T_{DNS2} + 3T_{F2H2} + T_{W2}$$

to go from the caller to the callee. The QoS request can only be initiated after the caller's CoA has been found, which takes

$$T_{CoA1} = 4T_{W1} + 4T_{W2} + 4T_{F2H2} + 3T_{H1H2} + 3T_{F1H1} + T_{F1H2}.$$

The QoS request/response at the callee side takes

$$T_{QoS1} = 2T_{W2}.$$

Then the 183 Session Progress takes

$$T_{SP} = T_{W1} + T_{W2} + T_{F2H2} + T_{F1H2}$$

to go from the callee to the caller. Finding out the callee's CoA takes

$$T_{CoA2} = T_{W1} + 4T_{W2} + 3T_{F1H2} + 3T_{F2H2} + T_{F1F2}.$$

The QoS request/response at the caller side takes

$$T_{QoS2} = 2T_{W1}.$$

Finally, the PRACK is sent to the callee with the SDP counter-answer, after which it may start ringing. Until the 180 Ringing arrives at the caller, there is an additional

$$T_{\Pr a} = 2T_{W1} + 2T_{W2} + T_{F1F2} + T_{F2H2} + T_{F1H2}.$$

Adding all of these delays, we get a dial-to-ringtone delay of

$$T_{Std} = 14T_{W1} + 14T_{W2} + 2T_{F1F2} + 5T_{F1H1} + 7T_{F1H2} + \\ + 12T_{F2H2} + T_{DNS1} + T_{DNS2}$$

7.2 Optimized Case

With our proposed optimizations (refer to fig. 3), the INVITE takes

$$T_{Inv} = T_{W1} + T_{DNS1} + T_{F1H2} + T_{F2H2} + T_{W2}$$

to go from the caller to the callee. Then, the QoS request takes

$$T_{QoS1} = 2T_{W2}.$$

Then the 183 Session Progress takes

$$T_{SP} = T_{W1} + T_{W2} + T_{F2H2} + T_{F1H2}$$

to go from the callee to the caller. The QoS request/response at the caller side takes

$$T_{QoS2} = 2T_{W1}.$$

Finally, the PRACK is sent to the callee, after which it may start ringing. Until the 180 Ringing arrives at the caller, there is an additional

$$T_{\Pr a} = 2T_{W1} + 2T_{W2} + T_{F1F2} + T_{F2H2} + T_{F1H2}.$$

Adding these delays, we get a dial-to-ringtone delay of

$$T_{Opt} = 6T_{W1} + 6T_{W2} + T_{F1F2} + 3T_{F1H2} + 3T_{F2H2} + T_{DNS1}$$

for the optimized signaling case.

If we consider $T_{W1} = T_{W2} = T_W$, $T_{DNS1} = T_{DNS2} = T_{DNS}$ and all inter-domain traversal delays equal to T_{ID}, then

$$T_{Std} = 28T_W + 26T_{ID} + 2T_{DNS}$$

$$T_{Opt} = 12T_W + 7T_{ID} + T_{DNS}$$

a more than twofold improvement.

8. Simulation results

The efficiency of the standard and optimized signaling scenarios was evaluated using the *ns-2* simulator [16] under Linux. The standard *ns-2* supports neither MIPv6 nor SIP. MIPv6 support was provided by the *mobiwan* extension [17], which we further improved by adding several features (reverse encapsulation, RRP, etc.). We have made an implementation of SIP, layered, with stateful entities, and supporting QoS-aware user agents (UA) and proxies/registrars; it also supports reliability of provisional responses (100rel) SIP extension [10], used in these simulations. This code is publicly available for download [18].

Some processing delays are accounted for in the simulation model. Message processing is performed in a FIFO fashion (processing of a message can only begin after all previous ones have been processed). Processing delays for SIP messages were simulated at both the terminals (15ms) and the MMSP (0.8ms), with an increment for messages with SDP bodies (10ms in the terminals and 0.8ms in the MMSP). QoS request processing at the QoS brokers is also accounted for (1ms). The remaining processing delays are considered negligible when compared to these ones, and thus ignored in the simulations. DNS lookups were not simulated for lack of a realistic model for DNS caching. Moreover, since our purpose is the evaluation of signaling, no actual session data was simulated.

Figure 4 shows the simulated topology. Though very simple, it allows us to simulate all possible combinations of roaming and non-roaming terminals. 128 terminals were uniformly spread among the access networks, each terminal having a 50% probability of being at its home domain and 50% of roaming. Random calls were generated between pairs of terminals,

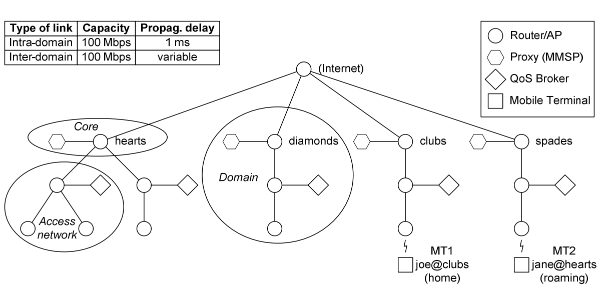

Figure 4. Topology used in the simulations

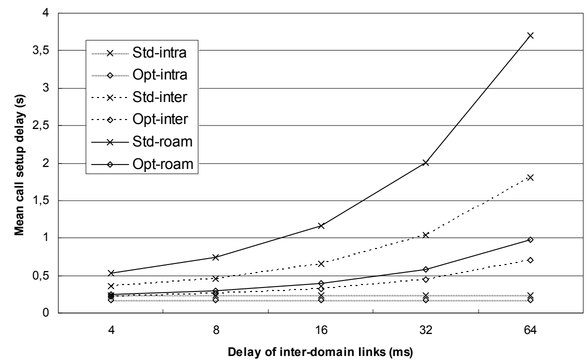

Figure 5. Call setup delay with varying inter-domain link propagation delay

with average duration of 120s and mean interval between generated call of 15s, for a simulated time of 24 hours (86400s). Several runs of each simulation were performed with different pseudo-random number generator (PRNG) seeds; different streams of the standard ns-2.27 PRNG were used for independent events.

In a first experiment we evaluated the call setup delay with different values of propagation delay in the inter-domain links. The setup delay is evaluated at the caller side, that is, from the moment the INVITE is sent to the moment the 200 OK for the INVITE is received and the ACK transmitted, subtracting the time it takes for the callee to answer the call (delay from sending the 183 Session Progress to sending the 200 OK). The results from this experiment are shown in fig. 5 for both the standard and optimized sequences, in three different scenarios: call within the same domain (*intra*), inter-domain call with both terminals at their home domains (*inter*) and inter-domain call with both terminals roaming (*roam*).

As expected, the setup delay does not vary with the propagation delay of inter-domain links in the *intra* scenario, since all signaling is performed intra-domain in this case. The worst scenario in terms of call setup delay is the *roam* (same as in figs. 2 and 3) — in this scenario, the difference in call setup delay between standard and optimized signaling is large, and increases with the propagation delay of inter-domain links. The 95% confidence intervals for the mean (5 runs) were less than ±3% of the mean in all cases.

In a second experiment we fixed the inter-domain propagation delay at 16 ms and introduced a varying loss probability at the wireless links; 802.11 MAC layer retransmissions were disabled so that losses were not compensated for.

The results of this experiment in different roaming scenarios, including 95% confidence intervals (10 runs) are shown in fig. 6. The figure clearly shows that the non-optimized scenario is much more severely affected by packet loss than the optimized one; this behavior stems from the much larger number of exchanged messages. It is worth noting that even with a packet loss ration of 1%, the mean setup delay of the most favorable roaming scenario (*intra*) with standard signaling was larger than that of the least favorable one (*roam*) with optimized signaling, a gap that is largely widened as the loss probability increases.

The above presented results show the clear advantage, in terms of call setup delay, of the optimized signaling method over the standard one. The improvement is even more dramatic for long distance calls (larger inter-domain propagation delays) and/or in the presence of packet loss in the wireless links, even though small.

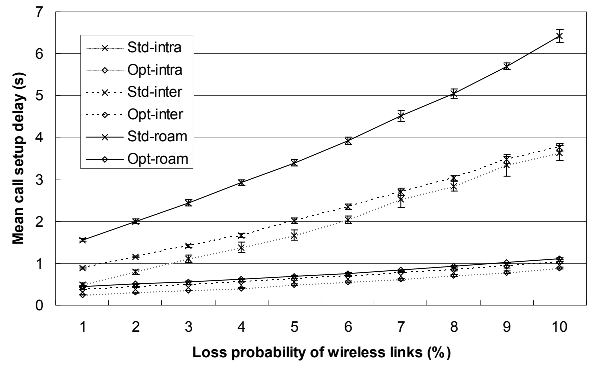

Figure 6. Call setup delay with varying loss probability in the wireless links

9. Conclusions

This paper identified the sources of inefficiency with the joint use of SIP and Mobile IPv6, particularly when end-to-end resource reservations must be performed for the media. This inefficiency generally stems from SIP/SDP's unawareness of layer 3 mobility, and from the need to perform resource reservations accounting for the physical points of attachment of the terminals. A solution for these inefficiencies was proposed, based on the direct use of the Care-of Addresses in some messages (namely for the short-lived message transactions in call initiation) and on a few cross-layer interactions, namely by including layer 3 location information in session setup signaling.

The advantages of the proposed optimizations in session establishment were analyzed, and simulation results have demonstrated that the session initiation sequence is much faster with the optimizations than in the standard case, particularly in the presence of larger inter-domain link propagation delays (long distance calls) or packet loss in the wireless links.

10. Acknowledgements

This work is based on results of the IST FP6 Integrated Project DAIDALOS, funded by the European Community's Sixth Framework Programme. It reflects the authors' views, and the EC is not liable for any use that may be made of the information herein contained.

11. References

[1] A. Banchs, I. Soto (eds.) et al., "Detailed Specifications including complete Interface Specifications." Daidalos consortium deliverable D221, Jan. 2005.

[2] M. Handley and V. Jacobson, "SDP: Session Description Protocol." IETF RFC 2327, Apr. 1998.

[3] D. Johnson, C. Perkins and J. Arkko, "Mobility Support in IPv6." IETF RFC 3775, Jun. 2004.

[4] J.W. Jung et al., "Mobility Agent with SIP Registrar for VoIP Services." LNCS, Vol. 2713, Jun. 2003.

[5] R. Koodli (ed.), "Fast Handovers for Mobile IPv6." IETF RFC 4068, Jul. 2005.

[6] D. Kutscher et al., "Session Description and Capability Negotiation." Internet Draft (draft-ietf-mmusic-sdpng-08), Feb. 2005.

[7] S. Olson et al, "Support for IPv6 in Session Description Protocol (SDP)." IETF RFC 3266, Jun. 2002.

[8] C. Politis, K.A.Chew, and R. Tafazolli, "Multilayer Mobility Management for All-IP Networks: Pure SIP vs. Hybrid SIP/Mobile IP." In Proc. VTC'2003 Spring, Apr. 2003.

[9] R. Prior, S. Sargento et al, "Providing End-to-End QoS in 4G Networks." In Proc. IASTED CCN'2005, Oct. 2005.

[10] J. Rosenberg and H. Schulzrinne, "Reliability of Provisional Responses in the Session Initiation Protocol (SIP)." IETF RFC 3262, Jun. 2002.

[11] J. Rosenberg et al., "SIP: Session Initiation Protocol." IETF RFC 3261, Jun. 2002.

[12] Q. Wang and M. Abu-Rgheff, "Integrated Mobile IP and SIP approach for advanced location management." In Proc. IEE 3G'2003, Jun. 2003.

[13] Q. Wang et al., "Design and Evaluation of an Integrated Mobile IP and SIP Framework for Advanced Handoff Management." In Proc. IEEE ICC'04, Jun. 2004.

[14] E. Wedlund and H. Schulzrinne, "Mobility Support using SIP." In Proc. ACM/IEEE WoWMoM'99, Aug. 1999.

[15] D. Willis and B. Hoeneisen, "Session Initiation Protocol (SIP) Extension Header Field for Registering Non-Adjacent Contacts." IETF RFC 3327, Dec. 2002.

[16] The Network Simulator – ns-2, version 2.27. http://www.isi.edu/nsnam/ns/

[17] Mobiwan MIPv6 extension to ns-2. http://www.ti-wmc.nl/mobiwan2/

[18] Ns-2 Network Simulator Extensions http://www.ncc.up.pt/~rprior/ns