

Home Agent Redundancy and Load Balancing in Mobile IPv6

F. Heissenhuber, W. Fritsche, and A. Riedl

IABG, Einsteinstr. 20, 85521 Munich, Germany

{heissenhuber;fritsche}@iabg.de

Lehrstuhl für Kommunikationsnetze, TU München, Arcisstr. 21, 80290 Munich, Germany

Anton.Riedl@ei.tum.de

Abstract This work proposes an optional extension to Mobile IPv6 to enhance the performance and the reliability of home agent clusters. First, a load balancing mechanism is presented which allows a set of home agents to distribute the load equally among them. Furthermore, we specify a Home Agent Redundancy Extension which increases the resilience of home agent clusters. In case of home agent failures, this extension enables the remaining peers to take over the functions of the missing home agents and, thus, keep up reachability of all currently registered mobile nodes. The extension is completely transparent for mobile nodes. The two mechanisms work independently of each other and, therefore, can be implemented separately. For better performance, though, it is advisable to apply both mechanisms at the same time.

Keywords: Mobile IPv6, Redundancy, Load Sharing

1. INTRODUCTION

Mobile IP will be an integral part of the next generation Internet Protocol [1,2]. It can be expected that the importance of mobility in the Internet will increase. Commercial providers could offer some kind of "Mobility Service" which allows their customers to go online at any point across the Internet and at the same time be reachable through their permanent home address.

In Mobile IPv6 each node is always identified by its home address, regardless of its current point of attachment to the Internet. While a mobile node is away from home it sends information about its current location to a home agent on its home link. The home agent stores the information about the remote location in its Binding Cache and advertises its own link-layer address for the IP address of the mobile node. Packets which are addressed to the mobile node's home address, are then intercepted by the home agent and tunneled to the mobile node's present location. After a certain time, a Binding Cache entry is timed out unless the mobile node refreshes it by sending a new Binding Update.

Home agents are single points of failure. If the home agent of a mobile node crashes, all communication to this mobile node which is routed over the home agent is disrupted. In Mobile IPv6, a mobile node discovers the loss of its home agent by sending registration messages periodically. When the mobile node does not receive acknowledgements, it performs Dynamic Home Agent Address Discovery to find a different suitable home agent. However, the time interval between registration requests is often set to be fairly large to avoid frequent traffic across the Internet. Thus, the most obvious way of detecting home agent failures may be too slow.

Although Mobile IPv6 allows several home agents on one link, it does not specify a redundancy mechanism in case one home agent fails. Our paper addresses this problem and proposes a Home Agent Redundancy Extension to enhance resilience. Furthermore, we describe a mechanism which enables load balancing among several home agents on one link. This becomes necessary when the number of Mobile IP users increases and the load cannot be handled by a single home agent anymore.

2. HOME AGENT LOAD BALANCING MECHANISM

In this work, we define load as the number of mobile nodes which are registered with a home agent. This load value is simple to determine and gives at the same time implicit information about some other load factors like the number of packets forwarded per time unit.

The load balancing mechanism utilizes Dynamic Home Agent Address Discovery which is initiated by a mobile host to find out the address of its home agent. When a mobile host is away from home it sends a Binding Update to the Home-Agents anycast address [3] of its home link. In response, one home agent on this link returns a list of all available home agents which are listed in order of decreasing preference value. The mobile

node selects one of the listed home agents and resends a Binding Update to this specific home agent. This completes the registration procedure.

Dynamic Home Agent Address Discovery requires every home agent to know the preference values of all other home agents. Therefore, the preference values are locally advertised by adding Home Agent Information options to Router Advertisements which are sent periodically. Preference values are specified as 16-bit signed, two-complement integers. As there is no standardized procedure to determine the Home Agent Preference value, we correlate it to the number of current registrations. Most preferable (32767) means 0 registrations. For every mobile host which is registered with a home agent, the preference value is decremented by one. Thus, the value -32768 (least preferable) represents 65535 bindings and more.

To realize load balancing, it is necessary that mobile nodes are not configured with a fixed IP address of their home agent. Instead, they must use Dynamic Home Agent Address Discovery to determine their home agent. The mobile node, upon receiving the Home Agents List, must try to register with each home agent in this list in the given order. Thus, it sends the first Binding Update to the home agent with currently the least number of registrations (= most preferable home agent). Normally, this home agent accepts the Binding Update, returns a Binding Acknowledgement, and decreases its preference value. Accordingly, when an entry in the Binding Cache of a home agent expires, a home agent increases its preference value.

Within one Router Advertisement period (0.5 to 1.5 seconds) the preference values in the Home Agents Lists of all home agents are updated, and a different home agent may become the most preferable one. Assuming a large number of mobile hosts, the load balancing mechanism should assure that all home agents have approximately the same number of registrations.

3. HOME AGENT REDUNDANCY EXTENSION

3.1 Definitions and New Messages

Rank: Every active home agent of a cluster is uniquely identified by its rank which is determined by the home agent's link-local address. The home agent with the lowest link-local address has rank 0. Higher link-local addresses result in higher ranks. When determining the rank at a certain time, only active home agents are considered.

All-home-agents multicast address: This is a multicast address which is assigned to all home agents using the Home Agent Redundancy Extension.

Home Agent Sequence Number: A home agent incrementally assigns a Home Agent (HA) Sequence Number to every accepted binding and forwards this number together with the binding information to all other home agents.

Extended Home Agents List: Every home agent maintains an Extended Home Agents List. It contains, among other information, a copy of the Binding Caches of all home agents which are compliant with the redundancy extension. The list is an extended version of the Home Agents List as described in [2].

Binding Update Forward Message: The Binding Update Forward message is used by a home agent to forward registration information about mobile nodes to the other home agents. It contains the Home Agent (HA) Sequence Number associated with the specific binding. The HA Sequence Number is used by home agents that receive a Binding Update Forward to detect missing Binding Update Forward messages.

Binding Cache Request Message: The Binding Cache Request message is used by a home agent to request the Binding Cache entries of another home agent.

Binding Cache Reply Message: The Binding Cache Reply message is sent in response to a Binding Cache Request. It contains the Binding Cache entries of home agents.

Home Agent Information Option: The Home Agent Information option is the same as specified in [2]. It is included in the Router Advertisements of a home agent. The Home Agent Redundancy Extension defines one additional bit, the Redundancy (R) bit. This bit is set by the sending home agent indicating that this home agent is compliant with the Home Agent Redundancy Extension.

3.2 Protocol Description

3.2.1 Redundancy Mechanism

When a home agent receives a Binding Update from a mobile node requesting home registration, the home agent processes it as usual. If it decides to reject the request, it sends a Binding Acknowledgement with the status field set accordingly. No further action takes place.

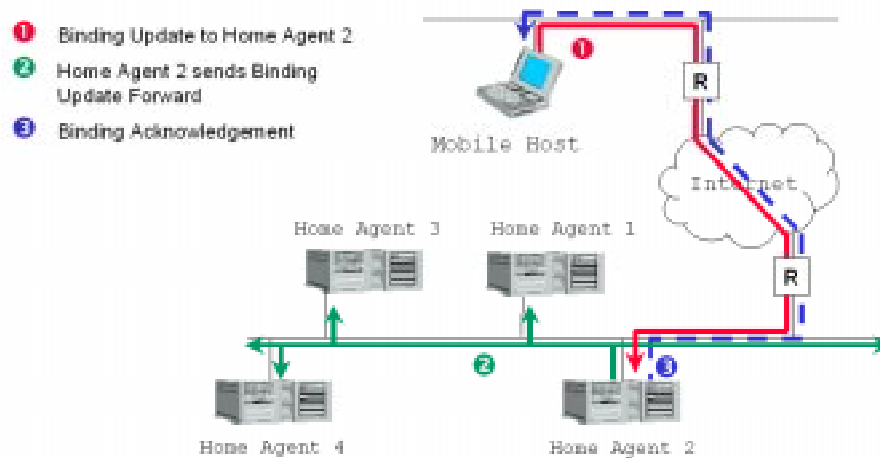


Figure 1. Forwarding Binding Updates

If, on the other hand, the home agent accepts the Binding Update, it stores information about the mobile node in its own Binding Cache and sends a Binding Update Forward message to the all-home-agents multicast address. This message contains the current address of the mobile host and a sequence number (HA Sequence Number) which is consecutively assigned to every accepted binding. Furthermore, the Update Forward Option carries a Lifetime and a Refresh field whose values are identical to the same fields in the respective Binding Acknowledgement. They indicate how long a binding will be valid until it has to be updated by the mobile host. All home registrations of a home agent are forwarded to the other home agents in the same way. Finally, to conclude the registration process the home agent sends a Binding Acknowledgement to the mobile node indicating successful registration (Figure 1). Sending a Binding Update Forward message for every mobile host registration enables all home agents to set up their Extended Home Agents Lists appropriately.

When a home agent receives a Binding Update Forward message, it first checks the HA Sequence Number whether it equals the expected value. If so, it searches all Binding Caches (also its own) in the Extended Home Agents List for an entry with the same home address as the new binding. If necessary, it deletes the old entry and adds the new binding to the Binding Cache of the home agent which has sent the Binding Update Forward message. It also stores the HA Sequence Number with this binding and starts a timer with the value that was given in the Lifetime field. If, however, an out-of-order HA Sequence Number indicates that one or more Binding Update Forward messages have not been received, the home agent tries to synchronize itself with the owner of the Binding Cache. Synchronizing the Extended Home Agents Lists at this point is similar to the Recovery Mechanism described in 3.2.3.

To reduce the traffic load on the home link, Binding Update Forward messages as well as Binding Cache Requests and Replies are sent by multicast. Since inconsistencies usually arise when Binding Update Forward messages are lost or the home link is divided, it is very likely that other home agents notice the same inconsistency. Therefore, Binding Cache Requests are sent to the all-home-agents multicast address. Moreover, the sending of this request is delayed by each home agent depending on its current rank. If a home agent receives a multicast Binding Cache Request while delaying the same request, it does not have to send this request itself. It just has to wait for the incoming multicast Binding Cache Reply. Using multicasting in this situation will reduce the number of packets on a large scale.

It should be noted that the entries of all Binding Caches are timed out independently. The deletion of a registration is not synchronized among the home agents. However, Binding Cache entries should time out at nearly the same time because multicast Binding Update Forward messages are received by all home agents almost simultaneously. Hence, the timer for the Binding Update is also started at almost the same time. Considering that the lifetime of a binding is assumed to be greater than one hour, the differences between the timers of different home agents for the same bindings are negligible.

3.2.2 Monitoring and Proxy Mechanism

A home agent which implements the Home Agent Redundancy Extension does not have to be configured with the addresses or names of the other home agents on a link. Every home agent listens to the Router Advertisements of all other home agents and uses this information to maintain the Home Agents List. A Router Advertisement which contains a Home Agent Information Option with the Redundancy (R) bit set indicates

that the sender is compliant with the Home Agent Redundancy Extension. Thus, every home agent is able to set up the Extended Home Agents List accordingly.

For monitoring the state of the peer home agents, no special messages are required. The monitoring mechanism is based on the periodic, unsolicited Router Advertisements. Every time a home agent receives a Router Advertisement of a peer home agent, it resets a timer. When a home agent fails to receive Router Advertisements from an already known peer for a certain amount of time (e.g., three times the Router Advertisement interval), it assumes that this home agent has been shut down or has crashed.

In case of a home agent failure, the remaining home agents act as proxies. They distribute the bindings of the respective Binding Cache among each other and take over the task of forwarding packets addressed to mobile hosts on behalf of the failed home agent. To determine which bindings are assigned to which home agent, every home agent performs a simple modulo computation for each entry in the Binding Cache. It is important to note that the home agents perform the calculation independently and that no additional messages are required. The calculation is based on the HA Sequence Number S which is stored with the binding and the number N of remaining active home agents. First, a home agent determines its current rank. Then, it computes $R = S \bmod N$ for every binding and assumes responsibility for this binding if R equals its rank.

The splitting of the bindings has been designed to be non-critical for unsynchronized expiration of bindings. Since the HA Sequence Numbers are unique within one Binding Cache, the modulo mechanism together with the rank of the home agents, guarantees an implicit and unequivocal distribution method. Furthermore, the monitoring mechanism ensures - even if several home agents fail - that every home agent is able to determine the correct rank of all home agents when it is necessary to do so.

3.2.3 Recovery Mechanism

When a home agent comes up, it immediately starts sending Router Advertisements without having set the Home Agent (H) bit. It starts a timer and listens for Router Advertisements which have the Redundancy (R) bit in the Home Agent Information Option set. If the timer expires and the home agent has not received such a Router Advertisement, it assumes that there is no other home agent (which is compliant with the Home Agent Redundancy Extension) on this link and starts without previous bindings.

However, if the home agent receives Router Advertisements which have the Redundancy (R) bit in the Home Agent Information Option set, it tries to synchronize its Binding Cache. To do so, it sends a Binding Cache Request

message (Figure 2). Binding Cache synchronization can be done with any home agent since all active home agents know the Binding Caches of all other home agents. If a home agent does not reply or if transmission fails, the home agent may choose to try it again or synchronize with a different home agent.

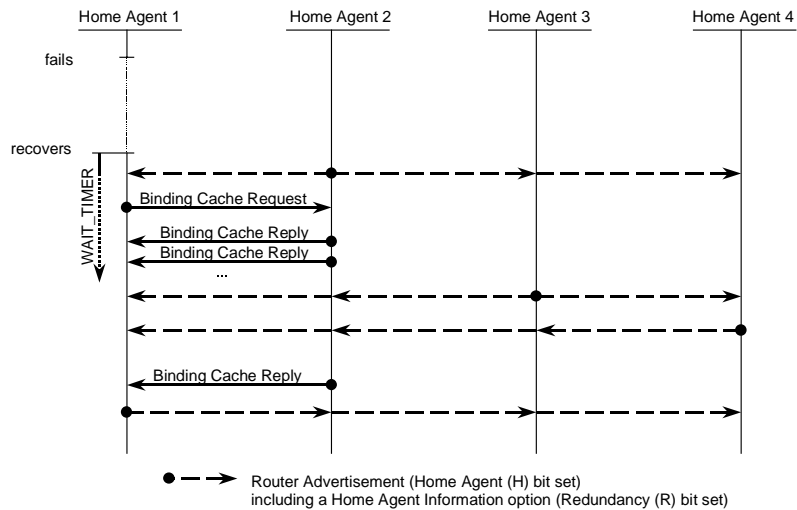


Figure 2. Data flow diagram: A home agent recovers

In case a home agent recovers after a crash, it also receives its own previous Binding Cache. For every entry of this Binding Cache, it immediately sends a Neighbor Advertisement to advertise its own link-layer address for the IP address of that mobile node.

When the home agent has completed the synchronization procedure, it sets the Home Agent (H) bit in the Router Advertisements and the Redundancy (R) bit in the Home Agent Information option. From now on, it is ready to accept registration requests and act as a proxy if another home agent fails.

3.3 Combination with Load Balancing

The Home Agent Redundancy Extension and the load balancing mechanism (Chapter 2) are independent of each other. It is very advisable, though, to use both mechanisms in parallel.

The load balancing mechanism allows the home agents to roughly share the load while doing regular operation. In case of failures, the Home Agent

Redundancy Extension ensures that the remaining home agents take over the task of forwarding packets.

Assuming a large number of mobile hosts, the distribution mechanism of the Home Agent Redundancy Extension should split the additional load equally among the remaining home agents. However, an unfavorable constellation could result in an unequal distribution of the bindings. In this case, the load balancing mechanism would restore a well-balanced situation.

3.4 Mobile Host Considerations

One of the main goals of this work is to require as little changes to the Mobile IPv6 protocol as possible. The Home Agent Redundancy Extension affects only the home agent's behavior and is completely transparent for mobile nodes. There is no need to make any changes to mobile nodes. Any Mobile IPv6 implementation for mobile nodes that is compliant with [1] is able to use the Home Agent Redundancy Extension without changes.

3.5 Bandwidth and Storage Requirements

Table 1 shows the messages which are used for redundancy and monitoring purpose. For every Binding Update which is accepted by a home agent, a Binding Forward and a Binding Acknowledgement has to be sent. Compared to standard Mobile IPv6 (only Binding Update + Binding Ack), the redundancy extension generates an additional load of 96 bytes (Binding Forward) for every Binding Update process.

Table 1. IP Packet Size

Message Type	IP packet size in bytes
Binding Update	104
Binding Acknowledgement	80
Binding Forward	96
Router Advertisement	96

However, the redundancy mechanism allows reducing the Binding Update frequency significantly. Home agent failures are remedied within three times the Router Advertisement interval (between 0.5 and 1.5 seconds by standard) independently of the Binding Update interval. To achieve similar fault tolerance in standard Mobile IPv6, Binding Updates would have to be sent at least every 4.5 seconds. Assuming a scenario with 10000 mobile hosts, this would create a load of about 3 Mbps across the Internet and on the local link. Choosing a Binding Update frequency of 1 hour could reduce this load to about 4 kbps across the Internet and 6kbps on the local link.

In case of a home agent failure, no extra load is created. Only when a home agent comes up after a failure, the redundancy extension generates additional messages (Binding Cache Request + Binding Cache Replies). The respective load which only appears on the local link depends on the number of registered mobile hosts.

The storage requirement when using the redundancy extension is somewhat higher than without the extension. A home agent needs to store information about every mobile host which is registered with any of the other home agents. This is not necessary in regular Mobile IPv6. However, the higher storage demand does not affect a home agent since a binding entry requires only little space (52 bytes with and 42 bytes without redundancy extension).

4. CONCLUSION

As Mobile IPv6 is implemented in commercial products, load balancing and fault tolerance become essential features of home agents. The current Mobile IPv6 specification leaves the employment of adequate mechanisms to the producers of network equipment. Therefore, we have presented two optional extensions to Mobile IPv6 which achieve load balancing and home agent redundancy, thus, enhancing the resilience of home agent clusters to a great extent. The two mechanisms are mainly based on already existing features of Mobile IPv6 and can be implemented without great effort. Only a few additional functions and messages are required. Although independent of each other, the complementary nature of the two mechanisms suggests their concurrent employment.

5. REFERENCE

1. S. Deering and R. Hinden, *Internet Protocol, Version 6 (IPv6) Specification*, RFC 2460, 1998
2. David B. Johnson and Charles Perkins, *Mobility Support in IPv6*, Internet Draft (work in progress), draft-ietf-mobileip-ipv6-08.txt, 1999
3. D. Johnson and S. Deering, *Reserved IPv6 Subnet Anycast Addresses*, RFC 2526, 1999