DMM Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 04, 2014

A. Yegin
K. Kweon
J. Lee
J. Park
Samsung

July 03, 2013

Corresponding Network Homing
draft-yegin-dmm-cnet-homing-00

Abstract

Mobile IP protocols provide IP session continuity to Mobile Nodes at the expense of creating triangular routes via a centralized Home Agent. Increased latency and network resource use, introduction of a single point of failure and a network choke point are among the undesirable side effects of the current protocols. This document describes an alternative approach where the Mobile Node makes use of dynamically-assigned Home Agent that is located close to the Corresponding Node.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

In the context of Mobile IP [RFC5563][RFC6275][RFC5213][RFC5944] following two attributes are defined for the IP service provided to the mobile hosts:

IP session continuity: The ability to maintain an ongoing IP session by keeping the same local end-point IP address throughout the session despite moving among different IP networks. The IP address of the host may change between two independent IP sessions, but that does not jeopardize the IP session continuity. IP session continuity is essential for mobile hosts to maintain ongoing IP sessions without any interruption.

IP address reachability: The ability to maintain the same IP address for an extended period of time. The IP address shall stay the same across independent IP sessions, and even in the absence of any IP session. The IP address may be published in a long-term registry (e.g., DNS), and it shall be available for serving incoming connections. IP address reachability is essential for mobile hosts to use specific/published IP addresses.

Mobile IP is designed to provide both IP session continuity and IP address reachability to mobile hosts. The basic operation of Mobile IP involves the following: Network assigning a fixed IP address to the Mobile Node (the Home Address, HoA) from a fixed node (the Home Agent, HA), the HA receiving location updates from the Mobile Node (MN), and the HA intercepting IP packets on behalf of the MN and tunneling them to the MN. That way the MN ensures it can keep receiving IP packets irrespective of its movement and location in the network.

One obvious side effect of this approach is the creation of sub-optimal routing paths between the MN and the other nodes it is communicating with (the Corresponding Nodes, CN). The routing path between the MN and a CN has to traverse the HA. Unless the HA is already located on the path between the MN and the CN, the path traversing the HA would create a so-called triangular route which is longer than the direct path between the two end-points. Longer path yields additional transmission latency and use of network resources [I-D.ietf-dmm-requirements].

Furthermore, forcing all MN traffic via the HA would also create a bottleneck in the network by overloading a single network element. The cost of building and operating such a network would increase, whereas the overall network reliability would decrease [I-D.ietf-dmm-requirements].

The objective of the solution described in this document is to provide IP session continuity to MNs without creating the aforementioned side effects.

The solution does not cover support for IP address reachability. This is considered to be acceptable, because only a very small set of applications really need IP address reachability. Those are the applications that are running as servers. Such applications cannot avoid using standard Mobile IP since they need to accept incoming connections at a specific/published IP address.

2. Notational Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. Solution in a Nutshell

The negative side effects of Mobile IP can be remedied if the HA were positioned on the direct path between the MN and the CN. That way the IP packets would naturally flow thru the HA and not follow a triangular route. But, this is not possible when a single/fixed IP address is assigned to the MN and it is served by a single HA at a fixed location [RFC5563][RFC6275][RFC5213][RFC5944] while the MN is communicating with CNs that are located in multiple different locations in the Internet.

The solution proposed in this document utilizes HAs located near CNs (Corresponding Home Agent, CHA) to dynamically allocate a HoA to the MN (Corresponding Home Address, CHoA). Such an address will be used throughout the IP session between the MN and the CN. Given the topological proximity of the CHA to the direct path between the MN and the CN, it is expected that this solution would not have the negative side effects of providing IP session continuity.

CHA may be co-located with the CN, or located in the same site as the CN, or located in an ISP serving that site. Not all CNs may be served by a CHA. In case there is no CHA serving the CN, the MN and the CN may communicate using the HoA via the HA. It is expected that CHAs would be deployed for dominant content sites on the Internet (e.g., YouTube, Facebook, Netflix, etc.)

The MN may be using multiple applications at the same time, and each application may be using a different CHA. For example, the MN may be configured to use CHoA1 for App1 with CN1 via CHA1, and use CHoA2 for App2 with CN2 via CHA2 at the same time.

Figure 1 depicts the high-level message flow for setting up data path between the MN and the CN using a CHA.

```
                    MN
                 +------+
                 |      |
                App   Stack      DNS      CHA     CN
                 |      |         |        |       |
                 |-[1]->|         |        |       |
                 |      |<--[2]-->|        |       |
                 |      |         |        |       |
                 |      |<----------[3]--->|       |
                 |      |         |        |       |
                 |      |===========[4]====|       |
                 |      |         |        |       |
                 |<-----============[5]====------->|
                 |      |         |        |       |
```

Figure 1. Data path setup.

Assume the MN is already configured with an IP address allocated from the access network it is attached to. Let this IP address be called IPxs.

Step 1:

Application on the MN attempts to initiate communication with the CN.

Step 2:

Network stack on the MN resolves the CN hostname to the IP address of CN. In parallel with that, the stack also tries to resolve the cha.CN_hostname in order to discover the CHA serving the CN, if there is any.

Step 3:

If a CHA is discovered at Step 2, then the network stack sends a Binding Update to the CHA. The HoA in the Binding Update is set to unspecified IP address (0.0.0.0/::) in order to request a dynamically-allocated CHoA from the CHA.

Step 4:

A tunnel is setup between the MN and the CHA as a result of Step 3. The tunnel end-points are IPxs and IP address of CHA (IPcha).

Step 5:

The socket used by the application is bound to the CHoA. The end-to-end communication between the App and the CN uses CHoA and IP address of CN (IPcn). Those IP packets are tunneled between the MN and the CHA.

Figure 2 depicts the high-level message flow for re-establishing the MN-CHA tunnel when the MN performs an IP handover.

```
                   MN
                    +------+
                    |      |
                   App   Stack               CHA     CN
                    |      |                  |       |
                    |     [1]                 |       |
                    |      |                  |       |
                    |      |<----------[2]--->|       |
                    |      |                  |       |
                    |      |===========[3]====|       |
                    |      |                  |       |
                    |<-----============[4]====------->|
                    |      |                  |       |
```

Figure 2. IP handover.

Step 1:

IP stack of MN configures a new IP address from the new access network (IPxs2).

Step 2:

MN sends a Binding Update to the CHA, binding CHoA to IPxs2.

Step 3:

A new tunnel is setup between the MN and the CHA (between the IPxs2 and IPcha).

Step 4:

The application and the CN continue their end-to-end communication using the new tunnel. The end point IP addresses stay the same (CHoA and IPcn), therefore the underlying routing change is transparent to the communication end-points.

Figure 3 depicts the high-level message flow for tearing down the MN-CHA tunnel when the application closes the connection.

```
                   MN
                    +------+
                    |      |
                   App   Stack               CHA     CN
                    |      |                  |       |
                    |-[1]->|                  |       |
                    |      |                  |       |
                    |      |<-=========[2]====------->|
                    |      |                  |       |
                    |      |<----------[3]--->|       |
                    |      |                  |       |
```

Figure 2. Tear down.

Step 1:

The application attempts to close the session with the CN.

Step 2:

The network stack closes the connection with the CN (e.g., TCP FIN).

Step 3:

The MN sends a Binding Update to the CHA in order to de-register the binding between the IPxs and the CHoA. Dynamically-allocated CHoA and the tunnel between the MN and the CHA are released at this step.

4. Details

This section provides a more detailed description of the proposed solution. The solution utilizes the standard Mobile IP protocol signaling. Unless otherwise stated, the protocol details in [RFC6275][RFC5944] apply to the Mobile IP processing described in the following sections.

4.1. Setup

Figure 4 depicts the detailed message flow for setting up data path between the MN and the CN using a CHA.

```
                    MN
                 +------+
                 |      |
                App   Stack      DNS      CHA     CN    AAA
                 |      |         |        |       |     |
                 |-[1]->|         |        |       |     |
                 |      |--[2a]-->|        |       |     |
                 |      |         |        |       |     |
                 |      |--[2b]-->|        |       |     |
                 |      |         |        |       |     |
                 |<-----|<-[3a]---|        |       |     |
                 |      |         |        |       |     |
                 |      |<-[3b]---|        |       |     |
                 |-[4]->|         |        |       |     |
                 |      |-----------[5]--->|       |     |
                 |      |         |        |--------[6]->|
                 |      |         |        |       |     |
                 |      |         |        |<-------[7]--|
                 |      |<----------[8]----|       |     |
                 |      |         |        |       |     |
                 |      |===========[9]====|       |     |
                 |      |         |        |       |     |
                 |      |<-=========[10]===------->|     |
                 |      |         |        |       |     |
                 |<-[11]-===================------>|     |
```

Figure 4. Detailed setup.

Step 1:

Application attempts to resolve the IP address of the CN by issuing a Socket API call (e.g., gethostbyname, getaddrinfo).

Step 2a:

DNS client on the MN sends a DNS request to the DNS server in order to resolve the IP address of the CN.

Step 2b:

In parallel with Step 2a, DNS client on the MN should also send a DNS request to the DNS server in order to resolve the IP address of cha.CN_hostname.

Steps 3a and 3b:

DNS server returns the results.

Step 4:

At some point, the application attempts to send its first packet to the CN by issuing a Socket API call (e.g., connect, sendto).

Step 5:

If Step 3b has produced an IP address for the CHA (which indicates availability of a CHA for the CN), then the MN shall send a Binding Update to the CHA. The Binding Update shall include a HoA that is set to the unspecified IP address (0.0.0.0 for IPv4, :: for IPv6).

Steps 6 and 7:

The CHA may need to authenticate the incoming Binding Update in order to authorize it. This step may require AAA [RFC2865][RFC3588] between the CHA and a AAA server.

Step 8:

The CHA shall return a Binding Acknowledgement to the MN. This message should contain a dynamically-allocated HoA. This HoA is regarded as a CHoA.

Step 9:

The MN shall configure the received CHoA on its stack. The MN and the CHA shall also setup a tunnel between the care-of address in the

Binding Update (the IPxs) and IPcha. The MN shall setup a routing table entry to forward any IP packet whose source address is CHoA to the CHA via the tunnel. The CHA shall setup a routing table entry to forward any IP packet whose destination address is CHoA to the MN via the tunnel.

Step 10:

The MN shall assign the newly-configured CHoA as the source address for the socket used by the application. If a connection-oriented transport protocol is used, then a connection shall be established between (e.g., via TCP 3-way handshake) the CHoA and the IPcn (as obtained in Step 3a) via the tunnel between the MN and the CHA.

Step 11:

The communication between the application and the CN shall use CHoA and the IPcn as the end-points, and go via the tunnel between the MN and the CHA.

4.2. Handover

Figure 5 depicts the detailed message flow for re-establishing the MN-CHA tunnel when the MN performs an IP handover.

```
                   MN
                    +------+
                    |      |
                   App   Stack               CHA
                    |      |                  |
                    |     [1]                 |
                    |      |                  |
                    |      |-----------[2]--->|
                    |      |                  |
                    |      |<----------[3]----|
                    |      |                  |
                    |      |===========[4]====|
```

Figure 5. Detailed handover.

Step 1:

The MN configures a new IP address from the new access network (IPxs2).

Step 2:

When the IP address of the MN changes, the MN shall send a Binding Update to the CHA for informing the CHA about this change and binding the CHoA to the new IP address. The Binding Update shall include a CoA set to the IPxs2 and the HoA set to the CHoA.

Step 3:

The CHA shall process the incoming Binding Update according to [RFC6275][RFC5944] and return a Binding Acknowledgement.

Step 4:

The MN and the CHA shall setup a new tunnel with each other upon successful execution of Steps 3 and 4. The tunnel end-points shall be set to IPxs2 and IPcha. The CHA shall forward any incoming packet whose destination is CHoA towards the MN via the tunnel. The MN shall forward any outgoing packet whose source address is CHoA towards the CN via the tunnel.

4.3. Teardown

Figure 6 depicts the detailed message flow for tearing down the MN-CHA tunnel when the application closes the connection.

```
                       MN
                    +------+
                    |      |
                   App   Stack              CHA     CN
                    |      |                  |       |
                    |-[1]->|                  |       |
                    |      |                  |       |
                    |      |<-=========[2]====------->|
                    |      |                  |       |
                    |      |-----------[3]--->|       |
                    |      |                  |       |
                    |      |<----------[4]----|       |
                    |      |                  |       |
```

Figure 6. Detailed tear down.

Step 1:

The application attempts to close the session with the CN.

Step 2:

The MN shall close the connection with the CN, if a connection-oriented transport is used (e.g., TCP, SCTP).

Step 3:

The MN shall send a Binding Update to CHA with lifetime set to 0.

Step 4:

The CHA shall send a Binding Acknowledgement back to the MN. The CHA and the MN shall release the tunnel, remove the forwarding entries for the CHoA. The CHA shall return the CHoA to the pool of available IP addresses. The MN shall unconfigure the CHoA on its stack.

If a connectionless protocol is used (e.g., UDP) between the MN and the CN, then the tear down may be triggered based on an inactivity timer or other indications.

5. Variation

A PMIP-based variation of this solution is under construction and will appear in a future version of this document.

6. Security Considerations

If the Binding Update message is not origin authenticated, then it may be leveraged for a DoS attack depleting the CHoA pool on the CHA.

This threat may be mitigated by allocating the CHoA from a very large address pool, such as 10.0.0.0/8 for IPv4, or a /64 prefix for IPv6. Depleting such a large address pool requires a significant brute-force attack. At that point the type of attack and its mitigations change and fall outside the scope of this document.

Another mitigation is to use origin authentication, replay and integrity protection on the Mobile IP messages [RFC6275][RFC5944][RFC4285] .

7. IANA Considerations

TBD

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC5944] Perkins, C., "IP Mobility Support for IPv4, Revised", RFC 5944, November 2010.

[RFC6275] Perkins, C., Johnson, D., and J. Arkko, "Mobility Support in IPv6", RFC 6275, July 2011.

8.2. Informative References

[I-D.ietf-dmm-requirements] Chan, A., Liu, D., Seite, P., Yokota, H., and J. Korhonen, "Requirements for Distributed Mobility Management", draft-ietf-dmm-requirements-05 (work in progress), June 2013.

[RFC2865] Rigney, C., Willens, S., Rubens, A., and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, June 2000.

[RFC3588] Calhoun, P., Loughney, J., Guttman, E., Zorn, G., and J. Arkko, "Diameter Base Protocol", RFC 3588, September 2003.

[RFC4285] Patel, A., Leung, K., Khalil, M., Akhtar, H., and K. Chowdhury, "Authentication Protocol for Mobile IPv6", RFC 4285, January 2006.

[RFC5213] Gundavelli, S., Leung, K., Devarapalli, V., Chowdhury, K., and B. Patil, "Proxy Mobile IPv6", RFC 5213, August 2008.

[RFC5563] Leung, K., Dommety, G., Yegani, P., and K. Chowdhury, "WiMAX Forum / 3GPP2 Proxy Mobile IPv4", RFC 5563, February 2010.

Authors’ Addresses

Alper Yegin
Samsung
Istanbul
Turkey

Email: alper.yegin@partner.samsung.com

Kisuk Kweon
Samsung
Suwon
South Korea

Email: kisuk.kweon@samsung.com

Jinsung Lee
Samsung
Suwon
South Korea

Email: js81.lee@samsung.com

Jungshin Park
Samsung
Suwon
South Korea

Email: shin02.park@samsung.com