Name-based Shim6: A name-based approach to host mobility

Zhongxing Ming
Dept. of Comp. Sci. & Tech., Tsinghua Univ., Tsinghua National Laboratory for Information Science and Technology

Javier Ubillos
Communication Networks and Systems laboratory
Swedish Institute of Computer Science

Mingwei Xu
Dept. of Comp. Sci. & Tech., Tsinghua Univ., Tsinghua National Laboratory for Information Science and Technology

ABSTRACT

The addressing scheme of today's Internet expects network nodes to be static and use a single interface. Communication flows bind to specific IP-addresses, making the IP-address act both as a topological locator as well as an end-point identifier. Mobile hosts break with this model , which make it difficult to maintain a connection during address change. A typical method to solve this problem is adding a layer of indirection by providing a name-space split between locators (IP-addresses) and end-point identifiers (names).

In this paper we introduce name-based sockets and apply a name-based approach to improve Shim6 (a host-based multihoming solution) to support host mobility. Name-based sockets do not introduce a new name-space split, but use the already deployed identity/locator-split scheme - domain names for identifiers and IP-addresses for locators. This makes the name-split fully backwards compatible and does not require new infrastructure. We show how a name-based approach intrinsically transforms a multihoming solution into a mobility solution and how the existing infrastructure is sufficient to provide the necessary signaling to support both unilateral and bilateral mobile scenarios. We evaluate the performance of the proposed approach in a lab environment and show that it achieves promising recovery time in both scenarios.

1. INTRODUCTION

Routing in IP-networks assumes a hierarchical scheme. Historically, this model has been successful, however this hierarchical model assumes that individual hosts or networks have a single point of attachment and do not move. Nodes and networks which do have multiple points of attachment (multihomed) or transient points of attachment (mobile) become, from the point of view of the network, separate instances of hosts/networks. The consequence is that a connection may not persist over multiple attachment points, or changing points. To provide address (locator) agility a host or network must either disseminate the change to the Internet [1] or receive a new IP-address which belongs to the address-range of the new network provider.

The root of this problem lies in the dual role of IP-addresses. An IP-address plays both the roles of a topological-locator as well as an end-system identifier. It is impossible for a host, which has multiple changing locators, to maintain a session over address changes without specialized application-layer libraries.

If a stationary host with multiple interfaces, e.g. a HDSPA connection and a Wi-Fi connection wishes to switch the active interface (multihoming), or if a mobile host changes networks, e.g. a laptop with a single wi-fi interface moves from one access point (AP) to another, the connection initiated using the original address must break down.

This is a well known problem-area with multiple proposed architectural solutions. However, the existing solutions either introduce new layers of indirections, e.g. triangular routing in MIP [2], or require new elements and/or requirements on the infrastructure, e.g. LISP [3].

In this paper we propose Name-based shim6 to solve host mobility problems in IP networks. Traditional shim6 [4] provides a multi-homing solution which does not require new infrastructure. Shim6 provides a shim-layer at the network-layer. It allows the application to bind to a pseudo IP-address which in turn can swap out with other addresses that reach the same end-point.

There are however, drawbacks using pseudo IP addresses. It introduces a new layer of indirection, which requires management. Also, an address in use in one session may have been assigned to a different host.

Introducing a name-based approach to shim6 receives several advantages: The name-based approach allows us to change the Shim6 semantics by replacing the pseudo IP-address with a name. This way, a host's locator becomes both practically and semantically decoupled from its identifier (name). Also, by moving the responsibility of resolving names (e.g. looking up a Fully Qualified Domain Name (FQDN) in Domain Name System (DNS)

) to the Operating System (OS), Shim6 may provide locator-agility at its discretion (e.g. shim6 functionality) without disturbing the upper layers.

The contribution of name based Shim6 is as follows: 1) We propose and implement name-based sockets as a socket-API abstraction so that an application developer may use names instead of IP-addresses. 2) We improve Shim6 and integrate it with name-based sockets to provide mobility and multihoming. 3) We evaluate the performance of the proposed mechanism using a real testbed.

2. THE NAME-BASED SHIM6 STRUCTURE

2.1 Shim6 mechanism

Shim6 [4] is a network layer solution to provide host multihoming, which was designed within the Internet Engineering Task Force (IETF). It provides a stable IP address for the application to identify the connection and keeps a list of locators available to the remote host. The communicating hosts continually exchange their available locators through which they can be reached as the locator set changes (in this paper, we use IP address and locator interchangeably). Shim6 uses REAchability Protocol (REAP) [5] to test the existing locator's reachability. However, as shim6 uses one of the locators as the Upper Layer Identifier (ULID), if that locator becomes invalid the connection MUST be dropped [4]. As a consequence, shim6 is not suitable for mobility as there might very well be periods when a host has zero-locators in its locator-list and the primary locator (the one used as ULID) is likely to become invalid during a move as well.

2.2 Name-based sockets

Name-based sockets (NBS) [6] provide a socket-API abstraction where an application developer may use names instead of IP-addresses.

Traditionally, the application does the resolution from a name (FQDN) to an IP-address and provides this address to the OS. This means that when the application calls the OS to open a connection, it opens a connection to a specific IP-address, not a specific host.

Using name-based sockets, the application only needs to manage the name, and binds its connections to names. This allows the OS to manage the addresses (locators) without disrupting the application. This results in the possibility of providing locator-agility, where the OS may change addresses during a communication for e.g. redundancy, performance and mobility reasons.

Name-based sockets provide a new address family (AF_NAME) for applications to use. This allows applications to open a socket to a remote name (FQDN). In conjunction to this, the address family AF_NAME also provides semantics to deal with names. For example, the `bind()` call allows the application to bind to a specific name rather than a specific IP-address.

2.3 Using NBS to improve Shim6

By using a name rather than a locator as the identifier that an application binds to, we can overcome the limitation that shim6 imposes when the locator used as ULID becomes invalid. The application binds to a name, and the OS is responsible for managing the locators. This change allows us to drop the primary-locator (the one used as ULID) and replace it with others. Figure 1 shows the concept.

Name-based shim6 also implements a new state called NO_LOC. This state allows the connection to be preserved during periods of complete disconnection, when the locator-list is empty.

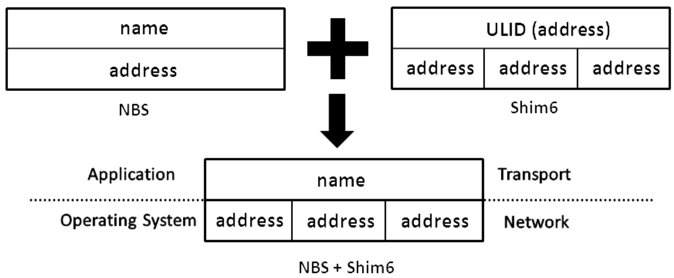

Figure 1: Name extension to traditional Shim6

2.4 Using DNS as a rendezvous point

The modifications described above work for the scenarios where a single host moves. In such scenarios, the moving host who has been disconnected still retains at least one live locator to the non-moving party to which it may send an updated locator-list. However, in the scenario where both hosts have switched their entire locator-set, there is no known locator to update to. To solve this problem, a rendezvous point is required. A rendezvous point is a well known point where clients may both register a locator on which they can be reached or query for a locator to a client they wish to communicate with. A good example is DNS. An user may query DNS for locators to a host of which the client knows the name, but does not know the locator. The locators may change over time, but the name remains constant. In a name-oriented architecture, it is natural to use DNS as a rendezvous point. If a host has no working locators to its correspondent party, it can query DNS for a new working locator [7].

3. IMPLEMENTATION AND EVALUATION

In this section we will investigate the performance of the proposed name-based Shim6 via a testbed implemented using household PCs.

3.1 Testbed implementation

An experimental implementation was developed under Ubuntu Linux operating system. NBS is developed as a standard Linux distribution. The topology of the testbed is shown in figure 2.

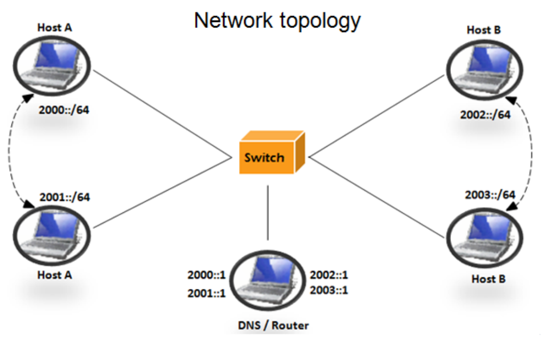

Figure 2: Testbed topology

As mobile event is just the same as renumbering event, during the connection, we use the 'ifconfig' command to artificially change the IP addresses of both hosts to simulate mobile events. To simulate concurrent move, we first delete one host's address and then delete that of another host, when both hosts loss their addresses, we reconfigure them with new addresses in arbitrary order.

The testbed was exhibited at IETF 79, Beijing, China.

3.2 Test results

To show the benefits of name based Shim6, we present the effect of a mobile event on the throughput of a TCP session in figure 3. A TCP session is established between host A and host B. During the connection, we first make one host move and then move both hosts simultaneously. We use wireshark to capture the throughput of the session.

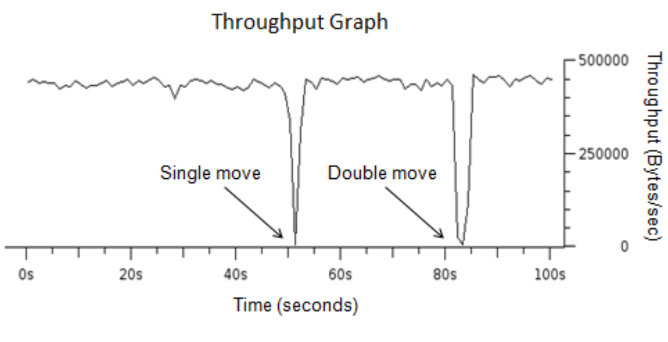

Figure 3: Throughput graph

In our experiment, host A moves from 2000::/64 to 2001::/64 at approximately 50 seconds after starting the wireshark client and both hosts move simultaneously at approximately 80 seconds.

The throughput drop represents the full recovery time, including the handover time during the mobile events. For single move scenario, the recovery time is about 1.65 seconds, and for double move scenario, the recovery time is approximately 2.75 seconds.

Hector Velayos and Gunnar Karlsson measured the handover time for different IEEE 802.11b cards [8]. They found that the handover times for the various cards are all around 1 - 2 seconds. This means that it is impossible to recover a connection in less than 1 - 2 seconds, which demonstrates that our mechanism achieves a reasonable recovery time.

4. CONCLUSION

This paper presents name-based shim6, a name-based improvement on a host-based multihoming solution to solve host mobility problems in IP networks.

We apply a name-based approach to improve Shim6 to support host mobility by integrating Shim6 with name-based sockets. We use DNS as a rendezvous point to coordinate address information for when both hosts move simultaneously. Using DNS makes the name/address-split fully backwards compatible and hence does not require new infrastructure. We evaluate the performance of the proposed approach by implementing the protocol and test it in a lab environment. Results show that our approach achieves promising recovery time when hosts move. Finding mechanisms to provide security guarantee for name resolution remains as a challenging issue for future research.

5. REFERENCES

[1] D. M. Vince Fuller. (2006, Mar.) Open issues with ipv6 routing / multihoming. [Online]. Available: http://www.ietf.org/proceedings/65/slides/grow-1.pdf

[2] C. Perkins, "Ip mobility support for ipv4," RFC3344, Internet Engineering Task Force, Aug 2002.

[3] D. Farinacci, V. Fuller, D. Meyer, and D. Lewis, "Locator/id separation protocol (lisp)," draft-ietf-lisp-11, Internet Engineering Task Force, Mar 2011.

[4] E. Nordmark and M. Bagnulo, "Shim6: Level 3 multihoming shim protocol for ipv6," RFC5533, June 2009.

[5] J. Arkko and I. van Beijnum, "Failure detection and locator pair," RFC 5534, June 2009.

[6] J. Ubillos, M. Xu, and Z. Ming, "Name based sockets," draft-ubillos-name-based-sockets-03, Sep 2010.

[7] L. Eggert, J. Laganier, L. M, and S. M, "Hip resolution and rendezvous mechanisms," 2004.

[8] H. Velayos and G. Karlsson, "Techniques to reduce the ieee 802.11b handoff time," in *IEEE International Conference on Communications*, 2004.