Network Working Group F. Templin, Ed.
Internet-Draft Boeing Research & Technology
Intended status: Informational September 29, 2017
Expires: April 2, 2018

IPv6 Prefix Delegation for Hosts
draft-templin-v6ops-pdhost-12.txt

Abstract

IPv6 prefixes are typically delegated to requesting routers which then use them to number their downstream-attached links and networks. This document considers the case when the requesting router is a node that acts as a host on behalf of its local applications and as a router on behalf of any downstream networks.

Table of Contents

1. Introduction

IPv6 Prefix Delegation (PD) entails 1) the communication of a prefix from a delegating router to a requesting router, 2) a representation of the prefix in the delegating router's routing table, and 3) a control messaging service between the delegating and requesting routers to maintain prefix lifetimes. Following delegation, the prefix is available for the requesting router's exclusive use and is not shared with any other nodes. An example IPv6 PD service is the Dynamic Host Configuration Protocol for IPv6 (DHCPv6) [RFC3315][RFC3633]. An alternative prefix management service based solely on IPv6 Neighbor Discovery (ND) messaging has also been proposed [I-D.pioxfolks-6man-pio-exclusive-bit].

This document considers the case when the requesting router is a node that acts as a host on behalf of its local applications and as a router on behalf of any downstream networks. The following paragraphs present possibilities for node behavior upon receipt of a delegated prefix.

For nodes that connect downstream-attached (aka "tethered") networks, a Delegating Router 'D' delegates a prefix 'P' to a Requesting node 'R' as shown in Figure 1:

```
                        +---------------------+
                        |Delegating Router 'D'|
                        |   (Delegate 'P')    |
                        +----------+----------+
                                   |
                                   | Upstream link
                                   |
                        +----------+----------+
                        |  Upstream Interface |
                        +---------------------+
                        |                     |
                        | Requesting node 'R' |
                        |    (Receive 'P')    |
                        |                     |
                        +--+-+--+-+--+-----+--+
                        |A1| |A2| |A3| ... |An|
                        +--+-+--+-+--+-----+--+
                        | Downstream Interface|
                        +----------+----------+
                                   |
                                   | Downstream link
                                   |
       X----+-------------+--------+----+---------------+---X
            |             |             |               |
       +---++-+--+   +---++-+--+   +---++-+--+     +---++-+--+
       |   |X1|  |   |   |X2|  |   |   |X3|  |     |   |Xn|  |
       |   +--+  |   |   +--+  |   |   +--+  |     |   +--+  |
       | Host H1 |   | Host H2 |   | Host H3 | ... | Host Hn |
       +---------+   +---------+   +---------+     +---------+

          <-------------- Tethered Network ------------->
```

Figure 1: Classic Routing Model

In this figure, when Delegating Router 'D' delegates prefix 'P', it inserts 'P' into its routing table with Requesting node 'R' as the next hop. Meanwhile, 'R' receives 'P' via an upstream interface and sub-delegates 'P' to its downstream external (physical) and/or internal (virtual) networks. 'R' assigns addresses 'A(i)' taken from 'P' to downstream interfaces, and Hosts 'H(i)' on downstream networks assign addresses 'X(i)' taken from 'P' to their interface attachments to the downstream link. 'R' then acts as a router between hosts 'H(i)' on downstream networks and correspondents reachable via other interfaces. 'R' can also act as a host on behalf of its local applications.

This document also considers the case when 'R' does not have any downstream interfaces, and can use 'P' solely for its own internal

addressing purposes. In that case, 'R' assigns 'P' to a virtual interface (e.g., a loopback) that fills the role of a downstream interface.

'R' can then function under the weak end system (aka "weak host") model [RFC1122][RFC8028] by assigning addresses taken from 'P' to a virtual interface as shown in Figure 2:

```
                     +---------------------+
                     |Delegating Router 'D'|
                     |   (Delegate 'P')    |
                     +----------+----------+
                                |
                                | Upstream link
                                |
                     +----------+----------+
                     |  Upstream Interface |
                     +---------------------+
                     |                     |
                     | Requesting node 'R' |
                     |    (Receive 'P')    |
                     |                     |
                     +--+-+--+-+--+-----+--+
                     |A1| |A2| |A3| ... |An|
                     +--+-+--+-+--+-----+--+
                     |  Virtual Interface  |
                     +---------------------+
```

Figure 2: Weak End System Model

'R' could instead function under the strong end system (aka "strong host") model [RFC1122][RFC8028] by assigning IPv6 addresses taken from 'P' to an upstream interface as shown in Figure 3:

```
                        +---------------------+
                        |Delegating Router 'D'|
                        |   (Delegate 'P')    |
                        +----------+----------+
                                   |
                                   | Upstream link
                                   |
                        +----------+----------+
                        |  Upstream Interface |
                        +--+-+--+-+--+-----+--+
                        |A1| |A2| |A3| ... |An|
                        +--+-+--+-+--+-----+--+
                        |                     |
                        | Requesting node 'R' |
                        |    (Receive 'P')    |
                        |                     |
                        +---------------------+
                        |   Virtual Interface |
                        +---------------------+
```

Figure 3: Strong End System Model

The major benefit for a node managing a delegated prefix in either the weak or strong end system models is multi-addressing. With multi-addressing, the node can configure an unlimited supply of addresses to make them available for local applications without requiring coordination with other nodes on upstream interfaces.

The following sections present considerations for nodes that employ prefix delegation mechanisms.

2. Terminology

The terminology of the normative references apply, and the terms "node", "host" and "router" are the same as defined in [RFC8200].

The following terms are defined for the purposes of this document:

shared prefix
   an IPv6 prefix that may be advertised to more than one node on the link, e.g., in a Router Advertisement (RA) message Prefix Information Option (PIO) [RFC4861].

individual prefix
   an IPv6 prefix that is advertised to exactly one node on the link, where the node may be unaware that the prefix is individual and may not participate in prefix maintenance procedures.

delegated prefix
an IPv6 prefix that is explicitly delegated to a node for its own exclusive use, where the node is an active participant in prefix delegation and maintenance procedures.

3. Multi-Addressing Considerations

IPv6 allows nodes to assign multiple addresses to a single interface. [RFC7934] discusses options for multi-addressing as well as use cases where multi-addressing may be desirable. Address configuration options for multi-addressing include StateLess Address AutoConfiguration (SLAAC) [RFC4862], DHCPv6 address configuration [RFC3315], manual configuration, etc.

Nodes configure addresses from a shared or individual prefix and assign them to the upstream interface over which the prefix was received. When the node assigns the addresses, it is required to use Multicast Listener Discovery (MLD) [RFC3810] to join the appropriate solicited-node multicast group(s) and to use the Duplicate Address Detection (DAD) algorithm [RFC4862] to ensure that no other node configures a duplicate address.

In contrast, a node that configures addresses from a delegated prefix can assign them without invoking MLD/DAD on an upstream interface, since the prefix has been delegated to the node for its own exclusive use and is not shared with any other nodes.

4. Multi-Addressing Alternatives for Delegated Prefixes

When a node receives a prefix delegation, it has many alternatives for provisioning the prefix. [RFC7278] discusses alternatives for provisioning a prefix obtained by a User Equipment (UE) device under the 3rd Generation Partnership Program (3GPP) service model. This document considers the more general case when the node receives a delegated prefix explicitly provided for its own exclusive use.

When the node receives the prefix, it can distribute the prefix to downstream networks and configure one or more addresses for itself on downstream interfaces. The node then acts as a router on behalf of its downstream networks and configures a default route via a neighbor on an upstream interface.

The node could instead (or in addition) use portions of the delegated prefix for its own multi-addressing purposes. In a first alternative, the node can assign as many addresses as it wants from the prefix to virtual interfaces. In that case, applications running on the node can use the addresses according to the weak end system model.

In a second alternative, the node can assign as many addresses as it wants from the prefix to the upstream interface over which the prefix was received. In that case, applications running on the node can use the addresses according to the strong end system model.

In both of these latter two cases, the node assigns the prefix itself to a virtual interface so that unused addresses from the prefix are correctly identified as unreachable. The node then acts as a host on behalf of its local applications even though neighbors on the upstream link see it as a router.

5. MLD/DAD Implications

When a node configures addresses for itself from a shared or individual prefix, it performs MLD/DAD by sending multicast messages over upstream interfaces to test whether there is another node on the link that configures a duplicate address. When there are many such addresses and/or many such nodes, this could result in substantial multicast traffic that affects all nodes on the link.

When a node configures addresses for itself from a delegated prefix, it can configure as many addresses as it wants but does not perform MLD/DAD for any of the addresses over upstream interfaces. This means that the node can configure arbitrarily many addresses without causing any multicast messaging over the upstream interface that could disturb other nodes.

6. Dynamic Routing Protocol Implications

The node can be configured to either participate or not participate in a dynamic routing protocol over the upstream interface, according to the deployment model. When there are many nodes on the upstream link, dynamic routing protocol participation might be impractical due to scaling limitations, and may also be exacerbated by factors such as node mobility.

Unless it participates in a dynamic routing protocol, the node initially has only a default route pointing to a neighbor via an upstream interface. This means that packets sent by the node over an upstream interface will initially go through a default router even if there is a better first-hop node on the link.

7. IPv6 Neighbor Discovery Implications

The node acts as a simple host to send Router Solicitation (RS) messages over upstream interfaces (i.e., the same as described in Section 4.2 of [RFC7084]) but also sets the "Router" flag to TRUE in its Neighbor Advertisement messages. The node considers the upstream

interfaces as non-advertising interfaces [RFC4861], i.e., it does not send RA messages over the upstream interfaces.

The current first-hop router may send a Redirect message that updates the node's neighbor cache so that future packets can use a better first-hop node on the link. The Redirect can apply either to a singleton destination address, or to an entire destination prefix as described in [I-D.templin-6man-rio-redirect].

8. ICMPv6 Implications

The Internet Control Message Protocol for IPv6 (ICMPv6) includes a set of control message types [RFC4443] including Destination Unreachable (DU).

According to [RFC4443], routers should return DU messages (subject to rate limiting) with code 0 ("No route to destination") when a packet arrives for which there is no matching entry in the routing table, and with code 3 ("Address unreachable") when the IPv6 destination address cannot be resolved.

According to [RFC4443], hosts should return DU messages (subject to rate limiting) with code 3 to internal applications when the IPv6 destination address cannot be resolved, and with code 4 ("Port unreachable") if the IPv6 destination address is one of its own addresses but the transport protocol has no listener.

Nodes that obtain and manage prefix delegations per this document observe the same procedures as described for both routers and hosts above.

9. IANA Considerations

This document introduces no IANA considerations.

10. Security Considerations

Security considerations for IPv6 Neighbor Discovery [RFC4861] and any applicable prefix delegation mechanisms apply to this document.

Additionally, the node may receive unwanted IPv6 packets via an upstream interface that match a delegated prefix but do not match either a configured IPv6 address or a transport listener. In that case, the node drops the packets and observes the "Destination Unreachable - Address/Port unreachable" procedures discussed in Section 8.

The node may also receive IPv6 packets via an upstream interface that do not match any of the node's delegated prefixes. In that case, the node drops the packets and observes the "Destination Unreachable - No route to destination" procedures discussed in Section 8. Dropping the packets is necessary to avoid a reflection attack that would cause the node to forward packets received from an upstream interface via the same or a different upstream interface.

In all cases, the node must decide whether or not to send DUs according to the specific operational scenario. In trusted networks, the node should send DU messages to provide useful information to potential correspondents. In untrusted networks, the node may refrain from sending DU messages to avoid providing sensitive information to potential attackers.

11. Acknowledgements

This work was motivated by discussions on the v6ops list. Mark Smith pointed out the need to consider MLD as well as DAD for the assignment of addresses to interfaces. Ricardo Pelaez-Negro, Edwin Cordeiro, Fred Baker, Naveen Lakshman, Ole Troan, Bob Hinden, Brian Carpenter, Joel Halpern and Albert Manfredi provided useful comments that have greatly improved the document.

This work is aligned with the NASA Safe Autonomous Systems Operation (SASO) program under NASA contract number NNA16BD84C.

This work is aligned with the FAA as per the SE2025 contract number DTFAWA-15-D-00030.

This work is aligned with the Boeing Information Technology (BIT) MobileNet program and the Boeing Research & Technology (BR&T) enterprise autonomy program.

12. References

12.1. Normative References

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, DOI 10.17487/RFC0791, September 1981, <https://www.rfc-editor.org/info/rfc791>.

[RFC3315] Droms, R., Ed., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, DOI 10.17487/RFC3315, July 2003, <https://www.rfc-editor.org/info/rfc3315>.

[RFC3633] Troan, O. and R. Droms, "IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6", RFC 3633, DOI 10.17487/RFC3633, December 2003, <https://www.rfc-editor.org/info/rfc3633>.

[RFC3810] Vida, R., Ed. and L. Costa, Ed., "Multicast Listener Discovery Version 2 (MLDv2) for IPv6", RFC 3810, DOI 10.17487/RFC3810, June 2004, <https://www.rfc-editor.org/info/rfc3810>.

[RFC4443] Conta, A., Deering, S., and M. Gupta, Ed., "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", STD 89, RFC 4443, DOI 10.17487/RFC4443, March 2006, <https://www.rfc-editor.org/info/rfc4443>.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, DOI 10.17487/RFC4861, September 2007, <https://www.rfc-editor.org/info/rfc4861>.

[RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, DOI 10.17487/RFC4862, September 2007, <https://www.rfc-editor.org/info/rfc4862>.

[RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <https://www.rfc-editor.org/info/rfc8200>.

12.2. Informative References

[I-D.pioxfolks-6man-pio-exclusive-bit] Kline, E. and M. Abrahamsson, "IPv6 Router Advertisement Prefix Information Option eXclusive Flag", draft-pioxfolks-6man-pio-exclusive-bit-02 (work in progress), March 2017.

[I-D.templin-6man-rio-redirect] Templin, F. and j. woodyatt, "Route Information Options in IPv6 Neighbor Discovery", draft-templin-6man-rio-redirect-04 (work in progress), August 2017.

[RFC1122] Braden, R., Ed., "Requirements for Internet Hosts - Communication Layers", STD 3, RFC 1122, DOI 10.17487/RFC1122, October 1989, <https://www.rfc-editor.org/info/rfc1122>.

[RFC7084] Singh, H., Beebee, W., Donley, C., and B. Stark, "Basic Requirements for IPv6 Customer Edge Routers", RFC 7084, DOI 10.17487/RFC7084, November 2013, <https://www.rfc-editor.org/info/rfc7084>.

[RFC7278] Byrne, C., Drown, D., and A. Vizdal, "Extending an IPv6 /64 Prefix from a Third Generation Partnership Project (3GPP) Mobile Interface to a LAN Link", RFC 7278, DOI 10.17487/RFC7278, June 2014, <https://www.rfc-editor.org/info/rfc7278>.

[RFC7934] Colitti, L., Cerf, V., Cheshire, S., and D. Schinazi, "Host Address Availability Recommendations", BCP 204, RFC 7934, DOI 10.17487/RFC7934, July 2016, <https://www.rfc-editor.org/info/rfc7934>.

[RFC8028] Baker, F. and B. Carpenter, "First-Hop Router Selection by Hosts in a Multi-Prefix Network", RFC 8028, DOI 10.17487/RFC8028, November 2016, <https://www.rfc-editor.org/info/rfc8028>.

Author's Address

Fred L. Templin (editor)
Boeing Research & Technology
P.O. Box 3707
Seattle, WA 98124
USA

Email: fltemplin@acm.org