        Problem Statement for Default Address Selection in Multi-Prefix
           Environments: Operational Issues of RFC 3484 Default Rules

Status of This Memo

Abstract

   A single physical link can have multiple prefixes assigned to it. In
   that environment, end hosts might have multiple IP addresses and be
   required to use them selectively. RFC 3484 defines default source
   and destination address selection rules and is implemented in a
   variety of OSs. But, it has been too difficult to use operationally
   for several reasons. In some environments where multiple prefixes
   are assigned on a single physical link, the host using the default
   address selection rules will experience some trouble in
   communication. This document describes the possible problems that
   end hosts could encounter in an environment with multiple prefixes.

Table of Contents

1. Introduction

   In IPv6, a single physical link can have multiple prefixes assigned
   to it. In such cases, an end host may have multiple IP addresses
   assigned to an interface on that link. In the IPv4-IPv6 dual-stack
   environment or in a site connected to both a Unique Local Address
   (ULA) [RFC4193] and globally routable networks, an end host typically
   has multiple IP addresses. These are examples of the networks that
   we focus on in this document. In such an environment, an end host
   may encounter some communication troubles.

   Inappropriate source address selection at the end host causes
   unexpected asymmetric routing, filtering by a router, or discarding
   of packets because there is no route to the host.

   Considering a multi-prefix environment, destination address selection
   is also important for correct or better communication establishment.

   RFC 3484 [RFC3484] defines default source and destination address
   selection algorithms and is implemented in a variety of OSs. But, it
   has been too difficult to use operationally for several reasons, such
   as lack of an autoconfiguration method. There are some problematic
   cases where the hosts using the default address selection rules
   encounter communication troubles.

   This document describes the possibilities of incorrect address
   selection that lead to dropping packets and communication failure.

1.1. Scope of This Document

   As other mechanisms already exist, the multi-homing techniques for
   achieving redundancy are basically out of our scope.

   We focus on an end-site network environment and unmanaged hosts in
   such an environment. This is because address selection behavior at
   these kinds of hosts is difficult to manipulate, owing to the users'
   lack of knowledge, hosts' location, or massiveness of the hosts.

   The scope of this document is to sort out problematic cases related
   to address selection. It includes problems that can be solved in the
   framework of RFC 3484 and problems that cannot. For the latter, RFC
   3484 might be modified to meet their needs, or another address
   selection solution might be necessary. For the former, an additional
   mechanism that mitigates the operational difficulty might be
   necessary.

   This document also includes simple solution analysis for each
   problematic case. This analysis basically just focuses on whether or
   not the case can be solved in the framework of RFC 3484. If not,
   some possible solutions are described. Even if a case can be solved
   in the framework of RFC 3484, as mentioned above, it does not
   necessarily mean that there is no operational difficulty. For
   example, in the environment stated above, it is not a feasible
   solution to configure each host's policy table by hand. So, for such
   a solution, the difficulty of configuration is yet another common
   problem.

2. Problem Statement

2.1. Source Address Selection

2.1.1. Multiple Routers on a Single Interface

```
                        ==================
                        |    Internet    |
                        ==================
                          |          |
          2001:db8:1000::/36 |          | 2001:db8:8000::/36
                   +----+-+      +-+----+
                   | ISP1 |      | ISP2 |
                   +----+-+      +-+----+
                        |          |
          2001:db8:1000:::/48 |          | 2001:db8:8000::/48
                   +-----+---+ +----+----+
                   | Router1 | | Router2 |
                   +-------+-+ +-+-------+
                         |          |
          2001:db8:1000:1::/64 |          | 2001:db8:8000:1::/64
                         |          |
                  -----+-+-----+------
                         |
                    +-+----+ 2001:db8:1000:1::100
                    | Host | 2001:db8:8000:1::100
                    +------+
```

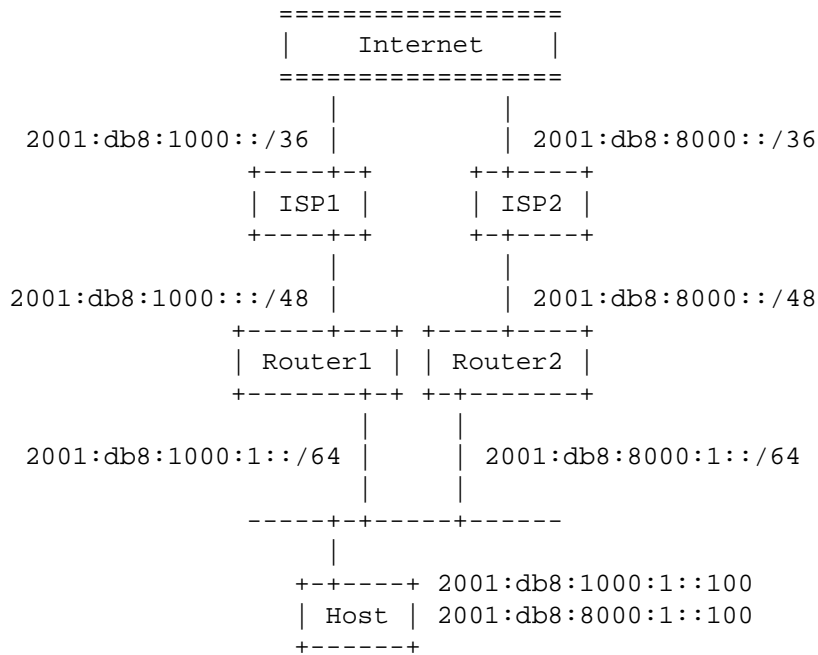   Figure 1

   Generally speaking, there is no interaction between next-hop
   determination and address selection. In this example, when a host
   starts a new connection and sends a packet via Router1, the host does
   not necessarily choose address 2001:db8:1000:1::100 given by Router1
   as the source address. This causes the same problem as described in
   the next section, "Ingress Filtering Problem".

   Solution analysis:
      As this case depends on next-hop selection, controlling the
      address selection behavior at the Host alone doesn't solve the
      entire problem. One possible solution for this case is adopting
      source-address-based routing at Router1 and Router2. Another
      solution may be using static routing at Router1, Router2, and the
      Host, and using the corresponding static address selection policy
      at the Host.

2.1.2. Ingress Filtering Problem

```
                  =================
                  |    Internet    |
                  =================
                        |        |
       2001:db8:1000::/36 |        | 2001:db8:8000::/36
                  +----+-+   +-+----+
                  | ISP1 |   | ISP2 |
                  +----+-+   +-+----+
                        |        |
       2001:db8:1000:::/48 |        | 2001:db8:8000::/48
                     ++-------++
                     | Router  |
                     +----+----+
                          |   2001:db8:1000:1::/64
                          |   2001:db8:8000:1::/64
                  ------+---+----------
                        |
                  +-+-----+ 2001:db8:1000:1::100
                  | Host  | 2001:db8:8000:1::100
                  +------+
```
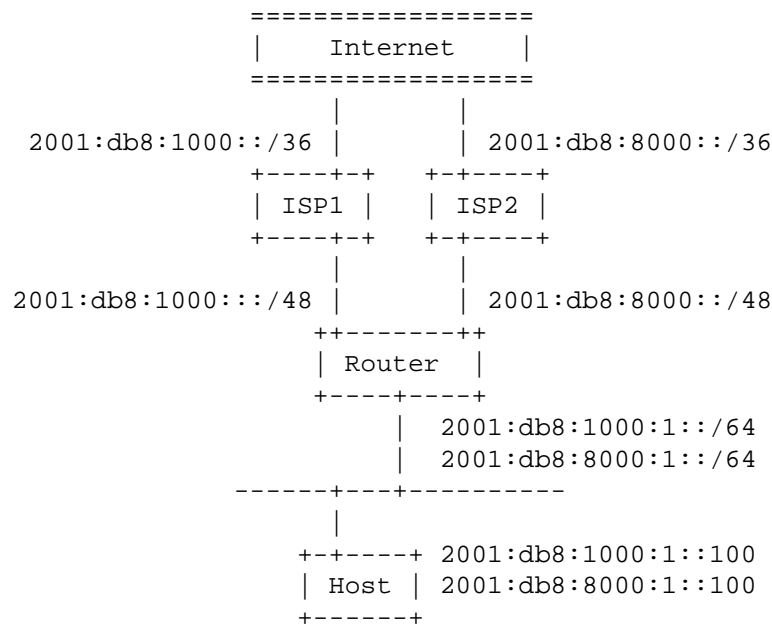
                              Figure 2

   When a relatively small site, which we call a "customer network", is
   attached to two upstream ISPs, each ISP delegates a network address
   block, which is usually /48, and a host has multiple IPv6 addresses.

   When the source address of an outgoing packet is not the one that is
   delegated by an upstream ISP, there is a possibility that the packet
   will be dropped at the ISP by its ingress filter. Ingress filtering
   is becoming more popular among ISPs to mitigate the damage of
   denial-of-service (DoS) attacks.

   In this example, when the router chooses the default route to ISP2
   and the host chooses 2001:db8:1000:1::100 as the source address for
   packets sent to a host (2001:db8:2000::1) somewhere on the Internet,
   the packets may be dropped at ISP2 because of ingress filtering.

   Solution analysis:
      One possible solution for this case is adopting source-address-
      based routing at the Router. Another solution may be using static
      routing at the Router, and using the corresponding static address
      selection policy at the Host.

2.1.3. Half-Closed Network Problem

   You can see a second typical source address selection problem in a
   multi-homed site with global half-closed connectivity, as shown in
   the figure below. In this case, Host-A is in a multi-homed network
   and has two IPv6 addresses, one delegated from each of the upstream
   ISPs. Note that ISP2 is a closed network and does not have
   connectivity to the Internet.

```
                      +--------+
                      | Host-C | 2001:db8:a000::1
                      +-----+--+
                            |
                =============  +--------+
                | Internet  |  | Host-B | 2001:db8:8000::1
                =============  +--------+
                    |            |
         2001:db8:1000:/36 |         | 2001:db8:8000::/36
                +----+-+   +-+---++
                | ISP1 |   | ISP2 | (Closed Network/VPN tunnel)
                +----+-+   +-+----+
                    |         |
         2001:db8:1000:/48 |     | 2001:db8:8000::/48
                     ++-------++
                     | Router  |
                     +----+----+
                          |   2001:db8:1000:1::/64
                          |   2001:db8:8000:1::/64
                ------+---+----------
                      |
                  +--+-----+ 2001:db8:1000:1::100
                  | Host-A | 2001:db8:8000:1::100
                  +--------+
```

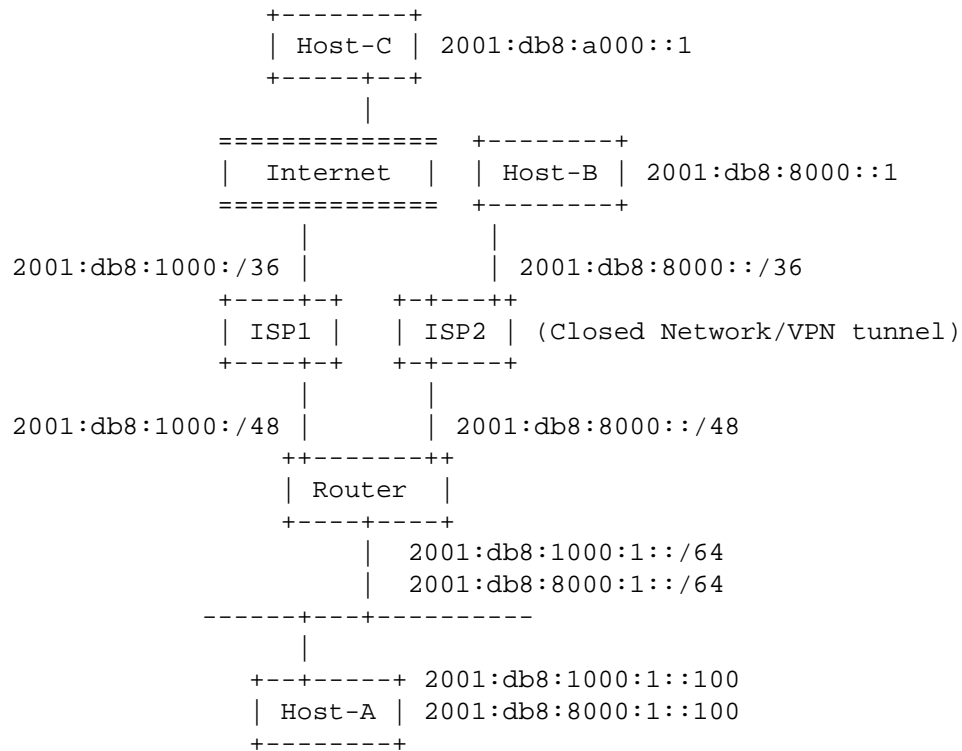   Figure 3

   You do not need two physical network connections here. The
   connection from the Router to ISP2 can be a logical link over ISP1
   and the Internet.

   When Host-A starts the connection to Host-B in ISP2, the source
   address of a packet that has been sent will be the one delegated from
   ISP2 (that is, 2001:db8:8000:1::100) because of rule 8 (longest
   matching prefix) in RFC 3484.

   Host-C is located somewhere on the Internet and has IPv6 address
   2001:db8:a000::1. When Host-A sends a packet to Host-C, the longest
   matching algorithm chooses 2001:db8:8000:1::100 for the source

address. In this case, the packet goes through ISP1 and may be
filtered by ISP1's ingress filter. Even if the packet is not
filtered by ISP1, a return packet from Host-C cannot possibly be
delivered to Host-A because the return packet is destined for 2001:
db8:8000:1::100, which is closed from the Internet.

The important point is that each host chooses a correct source
address for a given destination address. To solve this kind of
network-policy-based address selection problem, it is likely that
delivering additional information to a node provides a better
solution than using algorithms that are local to the node.

Solution analysis:
   This problem can be solved in the RFC 3484 framework. For
   example, configuring some address selection policies into Host-A's
   RFC 3484 policy table can solve this problem.
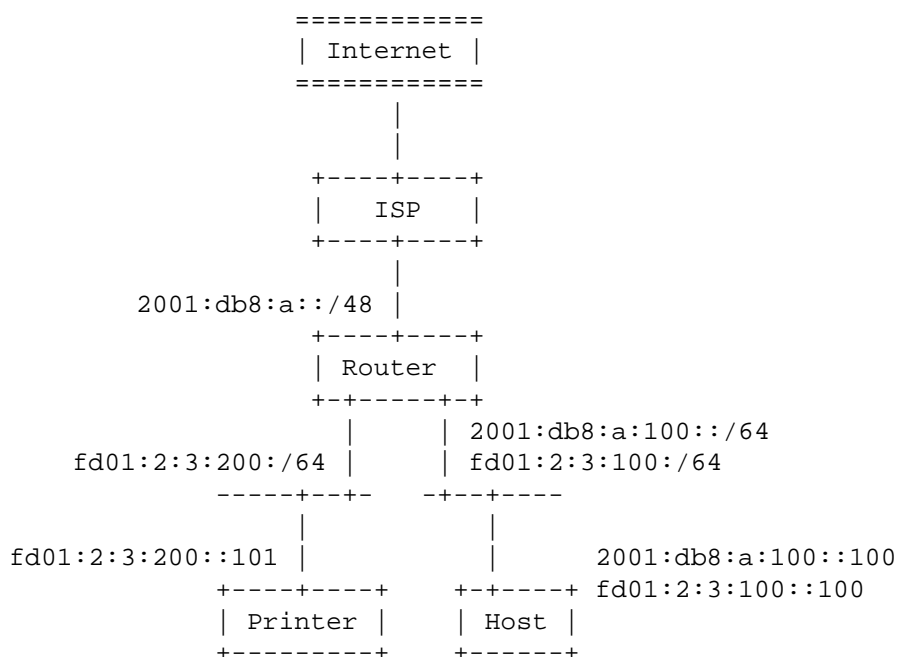
2.1.4. Combined Use of Global and ULA

```
                        ============
                        | Internet |
                        ============
                             |
                             |
                        +----+----+
                        |   ISP   |
                        +----+----+
                             |
               2001:db8:a::/48 |
                        +----+----+
                        | Router  |
                        +-+-----+-+
                          |     | 2001:db8:a:100::/64
          fd01:2:3:200:/64 |     | fd01:2:3:100:/64
               -----+--+-   -+--+----
                    |          |
       fd01:2:3:200::101 |          |        2001:db8:a:100::100
               +----+----+    +-+----+ fd01:2:3:100::100
               | Printer |    | Host |
               +---------+    +------+
```

                            Figure 4

As RFC 4864 [RFC4864] describes, using a ULA may be beneficial in
some scenarios. If the ULA is used for internal communication,
packets with the ULA need to be filtered at the Router.

   This case does not presently create an address selection problem
   because of the dissimilarity between the ULA and the global unicast
   address. The longest matching rule of RFC 3484 chooses the correct
   address for both intra-site and extra-site communication.

   In the future, however, there is a possibility that the longest
   matching rule will not be able to choose the correct address anymore.
   That is the moment when the assignment of those global unicast
   addresses starts, where the first bit is 1. In RFC 4291 [RFC4291],
   almost all address spaces of IPv6, including those whose first bit is
   1, are assigned as global unicast addresses.

   Namely, when we start to assign a part of the address block 8000::/1
   as the global unicast address and that part is used somewhere in the
   Internet, the longest matching rule ceases to function properly for
   the people trying to connect to the servers with those addresses.

   For example, when the destination host has an IPv6 address 8000::1,
   and the originating host has 2001:db8:a:100::100 and
   fd01:2:3:100::100, the source address will be fd01:2:3:100::100,
   because the longest matching bit length is 0 for 2001:db8:a:100::100
   and 1 for fd01:2:3:100::100, respectively.

   Solution analysis:
      This problem can be solved in the RFC 3484 framework. For
      example, configuring some address selection policies into the
      Host's RFC 3484 policy table can solve this problem. Another
      solution is to modify RFC 3484 and define ULA's scope smaller than
      the global scope.

2.1.5. Site Renumbering

   RFC 4192 [RFC4192] describes a recommended procedure for renumbering
   a network from one prefix to another. An autoconfigured address has
   a lifetime, so by stopping advertisement of the old prefix, the
   autoconfigured address is eventually invalidated.

   However, invalidating the old prefix takes a long time. You cannot
   stop routing to the old prefix as long as the old prefix is not
   removed from the host. This can be a tough issue for ISP network
   administrators.

   There is a technique of advertising the prefix with the preferred
   lifetime zero; however, RFC 4862 [RFC4862], Section 5.5.4, does not
   absolutely prohibit the use of a deprecated address for a new
   outgoing connection due to limitations on the capabilities of
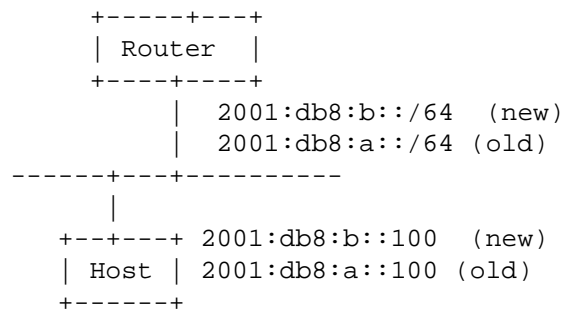   applications.

```
                          +-----+---+
                          | Router  |
                          +----+----+
                               |   2001:db8:b::/64  (new)
                               |   2001:db8:a::/64 (old)
                      ------+---+----------
                           |
                        +--+---+ 2001:db8:b::100  (new)
                        | Host | 2001:db8:a::100 (old)
                        +------+
```

                         Figure 5

   Solution analysis:
      This problem can be mitigated in the RFC 3484 framework. For
      example, configuring some address selection policies into the
      Host's RFC 3484 policy table can solve this problem.

2.1.6. Multicast Source Address Selection

   This case is an example of site-local or global unicast
   prioritization. When you send a multicast packet across site
   borders, the source address of the multicast packet should be a
   globally routable address. The longest matching algorithm, however,
   selects a ULA if the sending host has both a ULA and a Global Unicast
   Address.

   Solution analysis:
      This problem can be solved in the RFC 3484 framework. For
      example, configuring some address selection policies into the
      sending host's RFC 3484 policy table can solve this problem.

2.1.7. Temporary Address Selection

   RFC 3041 [RFC3041] defines a Temporary Address. The usage of a
   Temporary Address has both pros and cons. It is good for viewing web
   pages or communicating with the general public, but it is bad for a
   service that uses address-based authentication and for logging
   purposes.

   If you could turn the temporary address on and off, that would be
   better. If you could switch its usage per service (destination
   address), that would also be better. The same situation can be found
   when using an HA (home address) and a CoA (care-of address) in a
   Mobile IPv6 [RFC3775] network.

Section 6 ("Future Work") of RFC 3041 discusses that an API extension
might be necessary to achieve a better address selection mechanism
with finer granularity.

Solution analysis:
    This problem cannot be solved in the RFC 3484 framework. A
    possible solution is to make applications to select desirable
    addresses by using the IPv6 Socket API for Source Address
    Selection defined in RFC 5014 [RFC5014].

2.2. Destination Address Selection

2.2.1. IPv4 or IPv6 Prioritization

The default policy table gives IPv6 addresses higher precedence than
IPv4 addresses. There seem to be many cases, however, where network
administrators want to control the address selection policy of end
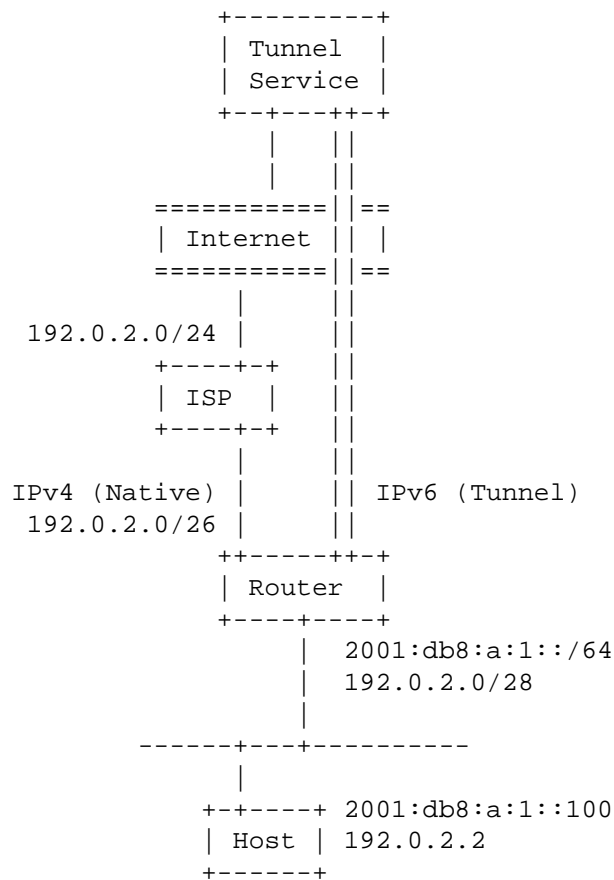hosts so that it is the other way around.

```
                           +---------+
                           | Tunnel  |
                           | Service |
                           +--+---++-+
                              |   ||
                              |   ||
                     ==========||==
                     | Internet ||  |
                     ==========||==
                         |     ||
            192.0.2.0/24 |     ||
                   +----+-+    ||
                   | ISP  |    ||
                   +----+-+    ||
                        |      ||
           IPv4 (Native) |     ||  IPv6 (Tunnel)
             192.0.2.0/26 |    ||
                   ++-----++-+
                   | Router  |
                   +----+----+
                        |   2001:db8:a:1::/64
                        |   192.0.2.0/28
                        |
             ------+---+----------
                   |
             +-+----+ 2001:db8:a:1::100
             | Host | 192.0.2.2
             +------+
```

                            Figure 6

   In the figure above, a site has native IPv4 and tunneled IPv6
   connectivity. Therefore, the administrator may want to set a higher
   priority for using IPv4 than for using IPv6 because the quality of
   the tunnel network seems to be worse than that of the native
   transport.

   Solution analysis:
      This problem can be solved in the RFC 3484 framework. For
      example, configuring some address selection policies into the
      Host's RFC 3484 policy table can solve this problem.

2.2.2. ULA and IPv4 Dual-Stack Environment

   This is a special form of IPv4 and IPv6 prioritization. When an
   enterprise has IPv4 Internet connectivity but does not yet have IPv6
   Internet connectivity, and the enterprise wants to provide site-local
   IPv6 connectivity, a ULA is the best choice for site-local IPv6

connectivity. Each employee host will have both an IPv4 global or
private address and a ULA. Here, when this host tries to connect to
Host-B that has registered both A and AAAA records in the DNS, the
host will choose AAAA as the destination address and the ULA for the
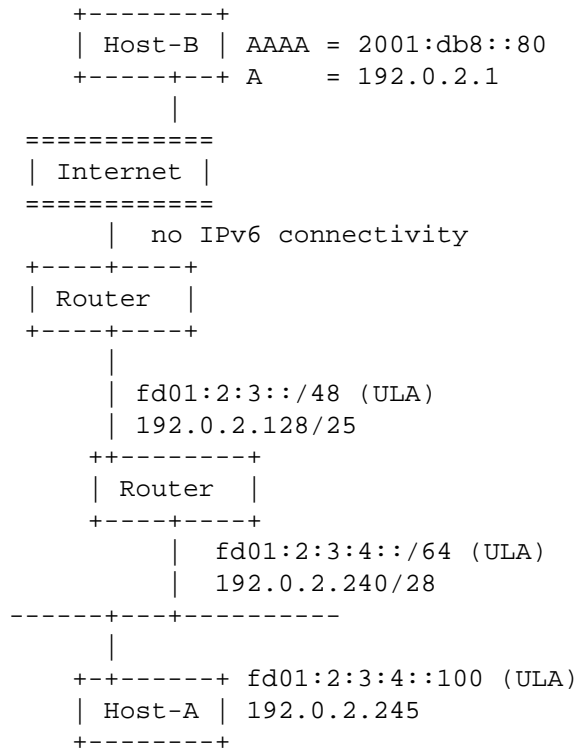source address. This will clearly result in a connection failure.

```
                    +--------+
                    | Host-B | AAAA = 2001:db8::80
                    +-----+--+ A    = 192.0.2.1
                          |
                 ===========
                 | Internet |
                 ===========
                      |  no IPv6 connectivity
              +----+----+
              | Router  |
              +----+----+
                    |
                    | fd01:2:3::/48 (ULA)
                    | 192.0.2.128/25
                 ++--------+
                 | Router  |
                 +----+----+
                       |   fd01:2:3:4::/64 (ULA)
                       |   192.0.2.240/28
              ------+---+----------
                    |
                +-+-------+ fd01:2:3:4::100 (ULA)
                | Host-A  | 192.0.2.245
                +--------+
```

                          Figure 7

   Solution analysis:
      This problem can be solved in the RFC 3484 framework. For
      example, configuring some address selection policies into Host-A's
      RFC 3484 policy table can solve this problem.

2.2.3. ULA or Global Prioritization

   Differentiating services by the client's source address is very
   common. IP-address-based authentication is a typical example of
   this. Another typical example is a web service that has pages for
   the public and internal pages for employees or involved parties. Yet
   another example is DNS zone splitting.

However, a ULA and an IPv6 global address both have global scope, and
RFC 3484 default rules do not specify which address should be given
priority. This point makes IPv6 implementation of address-based
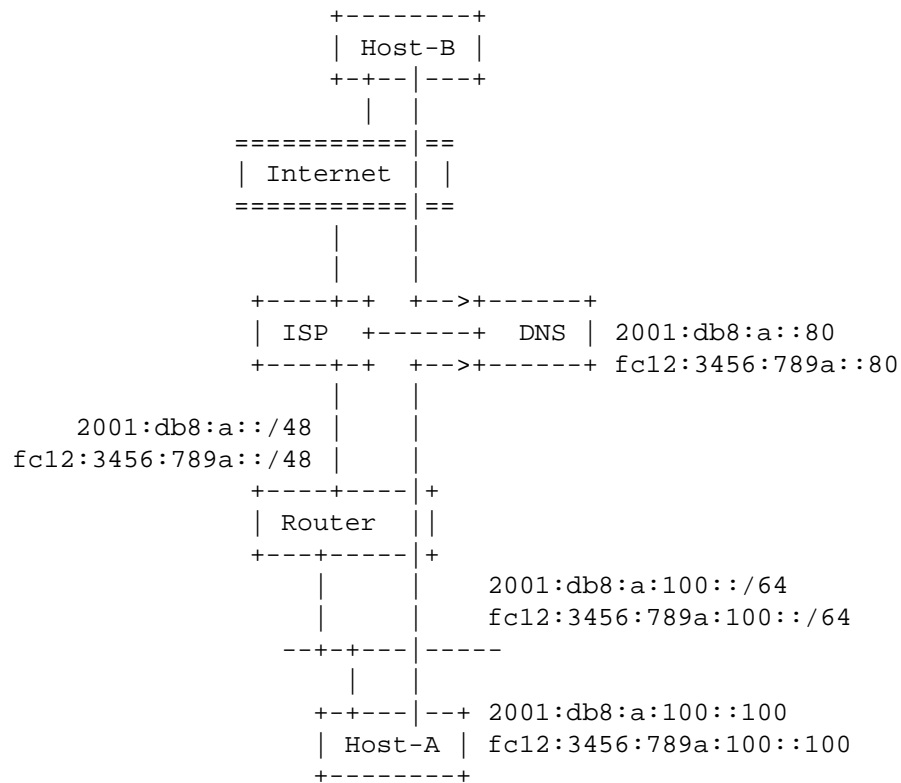service differentiation a bit harder.

```
                        +--------+
                        | Host-B |
                        +-+--|---+
                          |  |
                   ==========|==
                   | Internet | |
                   ==========|==
                          |  |
                          |  |
                   +----+-+  +-->+------+
                   | ISP  +------+  DNS | 2001:db8:a::80
                   +----+-+  +-->+------+ fc12:3456:789a::80
                        |    |
         2001:db8:a::/48 |    |
     fc12:3456:789a::/48 |    |
                   +----+----|+
                   | Router  ||
                   +---+-----|+
                       |     |     2001:db8:a:100::/64
                       |     |     fc12:3456:789a:100::/64
                    --+-+---|-----
                       |   |
                     +-+---|--+ 2001:db8:a:100::100
                     | Host-A | fc12:3456:789a:100::100
                     +--------+
```

                           Figure 8

   Solution analysis:
      This problem can be solved in the RFC 3484 framework. For
      example, configuring some address selection policies into Host-A's
      RFC 3484 policy table can solve this problem.

3. Conclusion

   We have covered problems related to destination or source address
   selection. These problems have their roots in the situation where
   end hosts have multiple IP addresses. In this situation, every end
   host must choose an appropriate destination and source address; this
   choice cannot be achieved only by routers.

It should be noted that end hosts must be informed about routing
policies of their upstream networks for appropriate address
selection. A site administrator must consider every possible address
false-selection problem and take countermeasures beforehand.

4. Security Considerations

When an intermediate router performs policy routing (e.g., source-
address-based routing), inappropriate address selection causes
unexpected routing. For example, in the network described in Section
2.1.3, when Host-A uses a default address selection policy and
chooses an inappropriate address, a packet sent to a VPN can be
delivered to a location via the Internet. This issue can lead to
packet eavesdropping or session hijack. However, sending the packet
back to the correct path from the attacker to the node is not easy,
so these two risks are not serious.

As documented in the Security Considerations section of RFC 3484,
address selection algorithms expose a potential privacy concern.
When a malicious host can make a target host perform address
selection (for example, by sending an anycast or multicast packet),
the malicious host can get knowledge of multiple addresses attached
to the target host. In a case like Section 2.1.4, if an attacker can
make the Host to send a multicast packet and the Host performs the
default address selection algorithm, the attacker may be able to
determine the ULAs attached to the host.

These security risks have roots in inappropriate address selection.
Therefore, if a countermeasure is taken, and hosts always select an
appropriate address that is suitable to a site's network structure
and routing, these risks can be avoided.

5. Normative References

   [RFC3041] Narten, T. and R. Draves, "Privacy Extensions for
              Stateless Address Autoconfiguration in IPv6", RFC 3041,
              January 2001.

   [RFC3484] Draves, R., "Default Address Selection for Internet
              Protocol version 6 (IPv6)", RFC 3484, February 2003.

   [RFC3775] Johnson, D., Perkins, C., and J. Arkko, "Mobility Support
              in IPv6", RFC 3775, June 2004.

   [RFC4192] Baker, F., Lear, E., and R. Droms, "Procedures for
              Renumbering an IPv6 Network without a Flag Day", RFC 4192,
              September 2005.

   [RFC4193] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast
              Addresses", RFC 4193, October 2005.

   [RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing
              Architecture", RFC 4291, February 2006.

   [RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless
              Address Autoconfiguration", RFC 4862, September 2007.

   [RFC4864] Van de Velde, G., Hain, T., Droms, R., Carpenter, B., and
              E. Klein, "Local Network Protection for IPv6", RFC 4864,
              May 2007.

   [RFC5014] Nordmark, E., Chakrabarti, S., and J. Laganier, "IPv6
              Socket API for Source Address Selection", RFC 5014,
              September 2007.

Authors' Addresses

   Arifumi Matsumoto
   NTT PF Lab
   Midori-Cho 3-9-11
   Musashino-shi, Tokyo 180-8585
   Japan

   Phone: +81 422 59 3334
   EMail: arifumi@nttv6.net

   Tomohiro Fujisaki
   NTT PF Lab
   Midori-Cho 3-9-11
   Musashino-shi, Tokyo 180-8585
   Japan

   Phone: +81 422 59 7351
   EMail: fujisaki@nttv6.net

   Ruri Hiromi
   Intec Netcore, Inc.
   Shinsuna 1-3-3
   Koto-ku, Tokyo 136-0075
   Japan

   Phone: +81 3 5665 5069
   EMail: hiromi@inetcore.com

   Ken-ichi Kanayama
   INTEC Systems Institute, Inc.
   Shimoshin-machi 5-33
   Toyama-shi, Toyama 930-0804
   Japan

   Phone: +81 76 444 8088
   EMail: kanayama_kenichi@intec-si.co.jp

Full Copyright Statement

Intellectual Property