Network Working Group
Internet-Draft
Expires: April 4, 2006

C. Huitema
R. Draves
Microsoft
M. Bagnulo
UC3M

October 2005

Ingress filtering compatibility for IPv6 multihomed sites
draft-huitema-shim6-ingress-filtering-00

Status of this Memo

Copyright Notice

Abstract

The note presents a set of mechanisms to provide ingress filtering compatibility for legacy hosts in IPv6 multihomed sites.

Table of Contents

1. Introduction

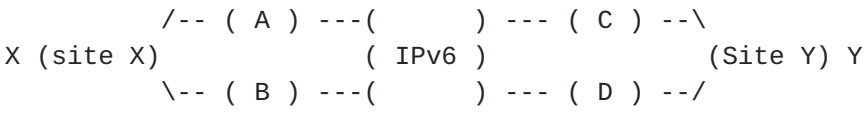
A way to solve the issue of site multihoming is to have a separate site prefix for each connection of the site, and to derive as many addresses for each hosts. This approach to multi-homing has the advantage of minimal impact on the inter-domain routing fabric, since each site prefix can be aggregated within the larger prefix of a specific provider; however, it opens a number of issues, that have to be addressed in order to provide a multihoming solution compatible with such addressing scheme.

In this memo we will present a set of mechanisms to deal with the problem created by the interaction between ingress filtering [3] and legacy hosts in multihomed sites.

The remaining of this memo is structured as follows: we will first present the reference topology and then the problem created by the interaction of ingress filtering and multihoming. Then, we will state the goals and non goals of the mechanisms presented in this memo. Next the proposed mechanisms are described. The Appendix A include a possible optimization for the case that the host within the multihomed site involved in the communication has special multihoming support.

2. Reference topology

In the following discussion, we will use this reference topology:



The topology features two hosts, X and Y, whose respective sites are both multi-homed. Host X has two global IPv6 addresses, which we will note "A:X" and "B:X", formed by combining the prefixes allocated by ISP A and B to "site X" with the host identifier of X. Similarly, Y has two addresses "C:Y" and "D:Y".

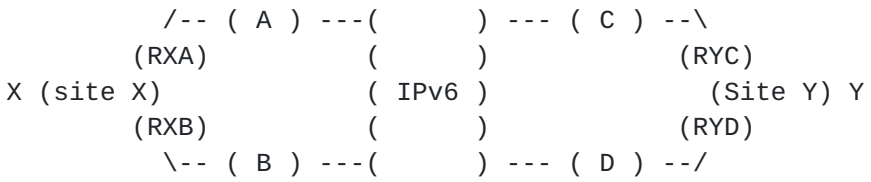
We assume that X, when it starts engaging communication with Y, has learned the addresses C:Y and D:Y, for example because they were published in the DNS. We assume that Y, when it receives packets from X, has only access to information contained in the packet coming from X, e.g. the source address; we do not assume that Y can retrieve by external means the set of addresses associated to X.

3. The problem: Ingress filtering incompatibility

Ingress filtering refers to the verification of the source address of the IP packets at the periphery of the Internet, typically at the link between a customer and an ISP. This process, which is described in [3] is intended to thwart a class of denial of service attacks in which attackers hide their identity by using a "spoofed" source address.

A special complication appears when the ISPs who serve the multihomed site perform ingress filtering. In the above configuration, X is served by ISP A and B, and thus can be reached by the addresses A:X and B:X. In addition Y is served by ISP C and D, and thus can be reached by the addresses C:Y and D:Y. To communicate with Y, X will choose the destination address that appears to be easier to reach, for example D:Y; then, it will choose the source address that provides the most efficient reverse path, say A:X.

Suppose now that the ISP connections at Site X are managed by two different site exit routers, RXA and RXB, and that there is a similar configuration at Site Y, with routers RYC and RYD.



Within Site X, the interior routing will decide which of RXA or RXB is the preferred exit router for the destination "D:Y"; similarly, within Site Y, the interior routing will decide which of RYC or RYD is the preferred exit for destination A:X. If the chosen exit router at Site X is RXA, the packet will flow freely to RYD; If the chosen exit router at Site Y is RYD, the response will also flow freely. However, if the exit routers are RXB or RYC, and if the ISPs perform ingress filtering, we have a problem: ISP B sees a packet coming from RXB, whose source address does not match the prefix assigned by B to X; ISP C, similarly, sees a packet whose source address does not match the prefix assigned by that ISP to Y. If either of these ISPs decides to drop the packet, the communication will be broken. Similar problems can also occur in communications between a host within a multihomed site and a host within a single-homed site.

4. Goals and non goals of the presented solution

4.1. Goal

The goal of the proposed solution is to provide ingress filtering compatibility for legacy hosts in multihomed environments, as described in RFC 3582 [2] .

"The solution should not destroy IPv6 connectivity for a legacy host implementing RFC 3513 [4] , RFC 2460 [1] , RFC 3493 [5], and other basic IPv6 specifications current in April 2003. That is to say, if a host can work in a single-homed site, it should still be able to work in a multihomed site, even if it cannot benefit from site-multihoming.

It would be compatible with this goal for such a host to lose connectivity if a site lost connectivity to one transit provider, despite the fact that other transit provider connections were still operational."[sic]

So, the goal of the presented solution is to enable a communication of two legacy hosts when at least one of them is in multihomed site, as long as no outage has occurred in the connectivity.

4.2. Non-goals

- It is not a goal of the presented solution to provide a complete multihoming solution. In particular, the presented solution does not provide fault tolerance capabilities (it does not preserve established communication through outages nor it enable hosts to initiate communications after an outage). So, the presented solution is a single component of a multihoming solution.

5. Proposed solution

In order to support legacy hosts, the addresses included in packets must be honored i.e. cannot be changed after that the host that is initiating the communication has selected them. So, there are two possible approaches to provide ingress filtering compatibility: to relax the ingress filtering or to perform some form of source address dependent routing. Both mechanisms will be presented next.

5.1. Relaxing the ingress filtering

An obvious way to avoid failures due to ingress filtering is to simply make sure that all the addresses used by the hosts of a given site will be considered acceptable by each of the site's providers. In our site X example, that would mean that provider A would accept addresses of the form "B:X" as valid, and that provider B will in turn accept addresses of the form "A:X" as valid.

One way to achieve this is simply to ask the service provider to turn off source address checks on the site connection. This requires a substantial amount of trust between the provider and the site, as source address checks are in effect delegated to the site routers. One possible way to achieve this trust is to make sure that the site routers, or possibly the site firewalls, meet a quality level specified by the provider.

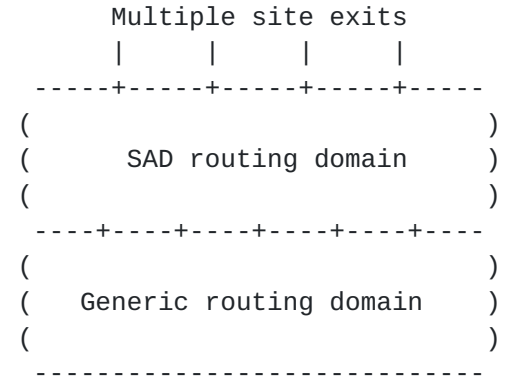
Another way to achieve this relaxed level of checking is to check source addresses against a list of "authorized prefixes" for the site connection, rather than simply the single prefix delegated by the provider. This solution requires that the site communicates the authorized prefixes to the provider, either through a management interface or through a routing protocol. This is obviously more complex than simply lifting the controls, and in fact ends up with a very similar requirement of trust: the provider has to be believe that the site will transmit the right prefixes.

In conclusion, relaxing the source address checks requires some form of explicit trust between the site and its providers. There is no doubt that this level of trust will exist in many cases; there is also no doubt that there will be many cases in which the provider is unwilling to grant this trust, particularly in the case of small sites, such as for example home networks dual-homes to a DSL provider and a cable network provider. So, this solution is a perfectly reasonable solution for large sites, i.e. the sites that benefit of IPv4 multihoming today: it should not be more complex to convince a provider to relax address checks for a particular customer tomorrow, than to convince today a similar provider to advertise in its routing table the global IPv4 address of the site. If we choose this

solution, we should choose its simplest implementation, i.e. one in which the provider completely delegates source address checks to the site's router or firewalls. This is however not a general solution, since we cannot expect all sites to convince every provider to relax their checks.

5.2. Source Address Dependent (SAD) routing

In order to provide ingress filtering compatibility it is possible to perform some kind of Source Address Dependent (SAD) routing within the site, so that the site exit is effectively a function of the source address in the packet. It should be noted that SAD routing does not have to be supported by all the routers of the multihomed site, but its support is only required to a connected domain that includes all the site exit routers as in the next figure:



In this schema, all site exit routers are connected to a SAD routing domain. Packets initiated in the generic routing domain and bound to an "out of site" address are passed to the nearest access point to the SAD routing domain, using classic "hot potato" routing. The routers in the SAD routing domain maintain as many parallel routing tables as there are valid source prefixes, and would choose a route that is a function of both the source and the destination address; the packets exit the site through the "right" router. There are multiple possible implementations of this general concept, depending on the site topology and the amount of routers involved. We will next present different scenarios and how SAD routing can be adopted in each of them.

5.2.1. Single site exit router

The simplest implementation is to have only one exit router for the site; in this case, the SAD routing domain is reduced to the exit router itself. This exit router chooses the exit link on the basis

of the source address in the packet. Many of the commercial routers already support this functionality so the solution in this cases could be easily adopted.

5.2.2. DMZ

A slightly less complex implementation is to connect all site exit routers to the same link, e.g. to what is often referred to as the "DMZ" for the site. This solution requires that all site exit routers connected to the DMZ support SAD routing. In this case, when a site exit router receives a packet, it verifies the source address; if the source address corresponds to its directly connected ISP, the site exit router forwards the packet through the ISP; if the source address of the packet does not corresponds to the directly connected ISP, the site exit router forwards the packet to the site exit router which ISP corresponds to the source address included in the packet.

In the case that a DMZ is not naturally available in the site, it is possible to create a virtual DMZ by using a mesh of tunnels between the site exit routers. In this case, a site exit router that receives a packet bound for an out-of-site address would perform a source address check before forwarding the packet on one of its outgoing interfaces; if the source address check is positive, the packet will effectively be sent on the interface; if it is not, the packet would be "tunneled" to the appropriate router.

The main requirement of the DMZ alternative is that site-exit routers be able to perform address checks, and that each site exit router be able to associate to each valid site prefix the address of a corresponding site exit router. An obvious possibility is to configure prefixes and corresponding addresses in each router; it would however be preferable to derive these addresses automatically. An assumption of the IPv6 architecture is that all prefixes of a site will have the same length; it is thus possible to derive a prefix from the source address of a "misdirected" packet, by combining this prefix with a conventional suffix. The suffix should be chosen to not collide with the subnet numbers used in the site; a null value will be inadequate, since it could be matched by any router with knowledge of the prefix, not just the site exit router; a value of "all ones" could be adequate.

So, each router managing a site prefix will then inject a "host route" announcing the anycast address associated to its locally managed prefixes in the interior routing protocol. Site exit routers can then use the standard routing procedures to detect whether the anycast address corresponding to the prefix in use is reachable; they can automatically reject, rather than forward, packets whose source address does not correspond to a reachable anycast address.

An inconvenience of this set-up is that some packet will follow a less than direct path; in the case of a natural DMZ, the additional path is probably negligible. However, in the case of a mesh of tunnels, the additional path may be significant, so this is not by itself a definitive solution. A possibility is to use this approach to support legacy hosts within the multihomed site and complement the solution by adopting the host based mechanism presented in Appendix A to allow upgraded host to select the optimal path. Another possibility is to use this approach as a way to "phase in" a full SAD routing solution described in the next section.

5.2.3. General case

In the most general set up, SAD routing is supported in all the site exit routers and all the internal routers required to create a connected SAD routing domain. Each router of the SAD domain would maintain as many parallel routing tables as there are valid source prefixes, and would choose a route that is a function of both the source and the destination address. Depending on how routes to external destinations are configured in routers the amount of work required to support SAD routing varies considerably.

5.2.3.1. Manual configuration

If routes to external destinations are manually configured, then the manual configuration of additional routes corresponding to each of the available prefixes is required. So, if within the multihomed site there are n prefixes and each router has m external routes configured, then (n-1)m additional routes need to be configured per router of the SAD routing domain.

It should be noted than multiple commercial routers currently support manually configured SAD routing, so this solution is already available.

It should also be noted that the usage of manually configured static routes does not preclude the fault tolerance capabilities of a multihoming solution, since fault tolerance is achieved by changing the prefix used for the communication, which is likely to be performed by the host itself, and not by the routing system. However, obtaining dynamic routing information may help the host to detect outages and enable faster response to outages.

5.2.3.2. Dynamic configuration

Information about external routes can be propagated within the multihomed site using a routing protocol, whether a IGP or BGP.

In order to support SAD routing in this scenario, routing protocols also have to convey information about the prefix associated with routing information that is being propagated. That is the prefix corresponding to the ISP through which the routing information was obtained.

When BGP is used, it would be possible to use a private community attribute to encode such information. However, current commercial routers don't support updating the different routing tables required to support SAD routing based on a BGP attribute (as far as we know).

In the general case, it is possible to run multiple instances of the routing protocol and associate each instance to a prefix, so that each instance of the routing protocol only carries information learned through the ISP corresponding to the prefix. However, our preliminary tests on this mechanisms seem to indicate that such mechanism wouldn't be currently supported by commercial routers.

It should be noted that having dynamic routing information about external routes does not provide fault tolerance to the multihomed sites as it did in IPv4-style multihoming, since, in sites with multiple prefixes, fault tolerance requires changing the prefix used for the communication. However, having dynamic routing information available does provide a faster feedback about failures, enabling faster response to outages.

6. Appendix A: Host based optimization

In this Appendix we present a host based mechanism to provide ingress filtering compatibility. The mechanism, called "Exit router discovery", enables the host to discover the preferred exit router for a given source address so that the host can tunnel the packet directly to the adequate exit router, obtaining optimal site exit path.

Exit router discovery is a natural complement of the tunneling mechanism between site exit routers. When an exit router tunnels a misdirected packet towards another exit, it may send an appropriate ICMP Destination Unreachable error message with code 5 which means source address failed ingress policy. If the host is a legacy host, the ICMP message will be ignored; further packets will continue using the same slightly sub-optimal path. On the other hand, if the host has been upgraded to take advantage of multi-homing, the packets will be tunneled to the appropriate exit router; they will follow a direct path to this router.

So, according to this mechanisms presented in this memo, site exit routers are expected to perform necessary source address checks before forwarding any packet on a site exit link. The amount of checking will vary depending on the exit link. If the provider has agreed to relax source address checking, the router will be configured to not do any checking at all; if the provider is expected to enforce a source address check, the site exit router must do the check first, in order to avoid local packets being routed to a black hole. If the result of the check is positive, the packet will be forwarded. If the result is negative, the router will derive a "site exit anycast address" from the source address of the incoming packet. If the anycast address is unreachable, the incoming packet will have to be discarded. If the anycast address is reachable, the incoming packet will be tunneled towards that address, and the router may issue a ICMP Destination Unreachable error message with code 5 which means source address failed ingress policy.

The reception of the ICMP packet is interpreted by the host implementing the exit discovery mechanism as an indication that the exit path being used is suboptimal. So, the host will first generate the site exit anycast address that corresponds to the source prefix of the packet that caused the generation of the ICMP packet (this is possible because the ICMP message payload contains enough information about the initial packet). Then, the host will associate this site exit anycast address to the source and destination address pair of the initial packet, and then tunnel packets directly to that exit router which will decapsulate these packets and send them over the appropriate exit link.

This mechanism requires a change to the caches used in neighbor discovery, specifically the management of a "source exit cache" that associates a specific source address with an exit router, or maybe the combination of a destination address and a source address with an exit router.

We should note that "exit router discovery" is not implemented in current hosts. We must meet the requirement expressed in RFC 3582 that hosts implementing the current version of IPv6 can continue to operate in a multi-homed site, even if they would not take advantage of multihoming; in consequence, these procedures can only be used as an optional optimization. It should also be noted that the presented mechanism does not requires any support from the host outside the multihomed site.

7. Security Considerations

There are elements presented in this draft that require further analysis from a security point of view: the usage of the anycast address of the site exit router associated to a given prefix and the usage of ICMP error messages to redirect following packets.

The usage of the anycast address of the site exit router router associated to a given prefix may enable potential attacks where the attacker announces the anycast address associated with a certain perifx and sinks the traffic containing such prefix in the source address. However, this threat is similar to the case where an attacker simply injects a route in the interior routing system, for instance a default route, sinking the packets of those routers and hosts that preffer such route.

An attacker could generate false ICMP errors to trigger the tunnelling of packets to the anycast address associated with the prefix of the source address. However, the result of such attack would simply be that packets are tunnelled through the appropriate site exit router. In the worst case, the packet would carry an extra tunnel header that would not be really required, causing additional overhead. In any case, these attack does not seem to cause cosniderable harm.

8. Acknowledgments

This memo contains parts of a previous work entitled "Host-Centric IPv6 Multihoming" that benefited from comments from Alberto Garcia Martinez, Cedric de Launois, Brian Carpenter, Dave Crocker, Xiaowei Yang and Erik Nordmark.

9. References

[1] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[2] Abley, J., Black, B., and V. Gill, "Goals for IPv6 Site-Multihoming Architectures", RFC 3582, August 2003.

[3] Ferguson, P. and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", RFC 2267, January 1998.

[4] Hinden, R. and S. Deering, "Internet Protocol Version 6 (IPv6) Addressing Architecture", RFC 3513, April 2003.

[5] Gilligan, R., Thomson, S., Bound, J., McCann, J., and W. Stevens, "Basic Socket Interface Extensions for IPv6", RFC 3493, March 2003.

Authors' Addresses

Christian Huitema
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052-6399
USA

Phone:
Email: huitema@microsoft.com
URI:

Richard Draves
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052-6399
USA

Phone:
Email: richdr@microsoft.com
URI:

Marcelo Bagnulo
Universidad Carlos III de Madrid
Av. Universidad 30
Leganes, Madrid 28911
SPAIN

Phone: 34 91 6249500
Email: marcelo@it.uc3m.es
URI: http://www.it.uc3m.es

Intellectual Property Statement

Disclaimer of Validity

Copyright Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.