Network Working Group	F. Baker
Internet-Draft	Cisco Systems
Intended status: BCP	November 08, 2007
Expires: May 11, 2008	

TOC

Source address validation in the local environment
draft-baker-sava-implementation-00

Status of this Memo

Abstract

This note describes how Source Address Validation might be applied in an IPv6 environment. Local systems should be able to ensure that their peers are using the IPv6 source addresses that the routing system uses to deliver data to them. Remote systems should be able to ensure that traffic they forward has reasonable source addresses.

Table of Contents

1. Introduction
- 1.1. Local Source Address Validation for IPv4
- 1.2. Local Source Address Validation for IPv6
- 1.3. Defense in depth

2. Implementating Source Address Validation in the local environment
- 2.1. Trust anchors
- 2.2. Host and Router validation of the addresses of neighboring

systems
    2.3. Validation of the addresses of neighboring systems by a switch
3. Implementating Source Address Validation for remote systems
4. IANA Considerations
5. Security Considerations
6. Contributors
7. References
    7.1. Normative References
    7.2. Informative References
Appendix A. Summary of recommendations
§ Author's Address
§ Intellectual Property and Copyright Statements

1. Introduction

TOC

This note describes how Source Address Validation might be applied in an IPv6 (Deering, S. and R. Hinden, “Internet Protocol, Version 6 (IPv6) Specification,” December 1998.) [RFC2460] environment by a an IP host or router, or lower layer switch directly connected to the source system. Local systems, the hosts and routers directly connected to a neighbor system, should be able to ensure that their peers are using the IPv6 source addresses that the routing system uses to deliver data to them. Remote systems cannot ensure the exact matching of addresses, but can determine whether the source address is reasonable. The recommendations of this specification are based on experience with such features that are currently available for IPv4 (Postel, J., “Internet Protocol,” September 1981.) [RFC0791] from multiple vendors.
This is in support of the requirements of [RFC2827] (Ferguson, P. and D. Senie, “Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing,” May 2000.), which recommends that networks defend themselves from spoofed traffic by dropping traffic that clearly has spoofed source addresses. The place this is usually implemented, the ISP ingress router, is an excellent second line of defense and very appropriate for defending the ISP. However, the only system that can definitively stop traffic from errant hosts is the systems they directly communicate with - their LAN switches, hosts that they are the immediate neighbors of, and their first hop routers. This note specifies that first line of defense.
The purpose of Source Address Validation is to ensure that a system in the network sends only datagrams that can be replied to - datagrams that the routing system will deliver to that host and which the host will recognize as having been directed to it. This is the first part is isolating a denial of service attack; the second step is to identify

systems sending attack traffic and remove their traffic from the network.

1.1. Local Source Address Validation for IPv4

TOC

In IPv4, datagrams generally come from the address that has been assigned to an interface, so it is sufficient to determine "the" address and discard traffic that comes from other addresses. There are some caveats: multi-LAN hosts may reply to a request using the address of one interface but sending the datagram out another, and routers forward traffic from a myriad of addresses. But such systems can be treated as special cases and excluded from source address validation. As such, switch implementations of source address validation technology observe DHCP (Droms, R., “Dynamic Host Configuration Protocol,” March 1997.) [RFC2131] assignments of addresses to hosts and drop host-originated datagrams using other source addresses. Neighboring hosts or routers may similarly only store one associated IP address for each MAC address, but this is more frequently a bug or side-effect of implementation than something well thought through, and may not operate as well as one would like.

1.2. Local Source Address Validation for IPv6

TOC

In an IPv6 network, the problem is somewhat more complicated than it is in an IPv4 network. As with IPv4, there are some caveats: multi-LAN hosts may reply to a request using the address of one interface but sending the datagram out another, and routers forward traffic from a myriad of addresses. But such systems can be treated as special cases and excluded from source address validation.

The issue that complicates IPv6 is that each interface potentially has many addresses, and a single prefix may straddle multiple physical interfaces. It will generally have at least two: its link-local address and its address in the local prefix. There may, however, be multiple local prefixes, and with privacy addressing (Narten, T., Draves, R., and S. Krishnan, “Privacy Extensions for Stateless Address Autoconfiguration in IPv6,” September 2007.) [RFC4941] there may be multiple legitimate addresses within the same prefix. So the mapping is not one-to-one, but rather one-to-many. The system verifying the address usage must carry the interface identification, in whatever form it may hold that, as an attribute of the IPv6 address, and verify that a datagram using a given IPv6 source address comes from the appropriate source.

1.3. Defense in depth

TOC

There is also a place for ingress filtering by prefix, usually accomplished via Unicast Reverse Path Forwarding or via filters. In general, this is performed between administrations - at the interface between peer ISPs, an ISP and its customer, or between two networks of another type. In concept, however, it could be applied on every router in a network. This will be discussed in Section 3 (Implementating Source Address Validation for remote systems).

2. Implementating Source Address Validation in the local environment

TOC

This section will describe in general terms a common approach to source address validation between two directly connected systems.
Addresses are allocated to systems in two ways in IPv6: DHCP (Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, “Dynamic Host Configuration Protocol for IPv6 (DHCPv6),” July 2003.) [RFC3315], Stateless Address Autoconfiguration (Thomson, S., Narten, T., and T. Jinmei, “IPv6 Stateless Address Autoconfiguration,” September 2007.) [RFC4862]. These addresses are exchanged with a system's neighbors using either Neighbor Discovery (Narten, T., Nordmark, E., Simpson, W., and H. Soliman, “Neighbor Discovery for IP version 6 (IPv6),” September 2007.) [RFC4861], or for Cryptographically Generated Addresses (Aura, T., “Cryptographically Generated Addresses (CGA),” March 2005.) [RFC3972], SEcure Neighbor Discovery (Arkko, J., Kempf, J., Zill, B., and P. Nikander, “SEcure Neighbor Discovery (SEND),” March 2005.) [RFC3971]. Each of these will be looked at.

2.1. Trust anchors

TOC

In short, the implementation approach requires neighboring devices to associate a layer 3 entity addressed with an IPv6 address with a layer 2 entity. The identification of the layer 2 entity may take a number of forms:

*On a traditional Ethernet or for a host or router attached to a switched Ethernet, the only real option is association with the MAC Address.

*The switch, however, may be able to associate the IPv6 address with a switch port if it knows that only one host is attached to the port.

*In a wireless network, there are often layer 2 security associations between neighboring devices, and the address can be associated with that security association.

*In a Cable Modem network, the address may be associated with the combination of a MAC address and a customer relationship.

*In a classical DSL network, it may be associated with an ATM Virtual Channel, or a PPPoE or L2TP Session ID.

*In some cases, an IP/IP tunnel, an MPLS LSP, or similar tunneling technology is taken to a single system. In such cases, local address validation can be applied to the use of the tunnel.

The key is to have a solid understanding of

*what identifies a neighbor in the context: a MAC Address, a switch port, an ATM VC, or whatever,

*what addresses one's neighbor is in fact using, and

*to limit what one accepts from the neighbor as so identified to that which the neighbor is legitimately using.

One could argue that this runs counter to the Robustness Principle, which is stated in RFC 793 as "be conservative in what you do, be liberal in what you accept from others." In fact, it follows the Robustness Principle, but adds the Russian proverb made famous in the west by Ronald Reagan: "Trust, but verify".

2.2. Host and Router validation of the addresses of neighboring systems

TOC

Since a peer's neighbors are intended to learn its address using Neighbor Discovery or Secure Neighbor Discovery, they should look there. If one system sends a datagram to a host or router on the same LAN or otherwise directly connected and the receiving system does not know the address to have been associated with the system that sent it, both of these protocols ask the receiver to query the sender using a Neighbor Solicit. Generally, this is done when sending the reply - the Neighbor Solicit and responding Neighbor Advertisement must be exchanged to send the application reply.
This specification recommends that, to protect hosts from attack traffic and prevent routers from forwarding datagrams with spoofed addresses, the NS/NA exchange SHOULD happen before the received datagram is operated on.

There are two schools of thought on the holding of the datagram; one holds that the host or router should hold the datagram in a short queue and release it on receipt of the NA, and one holds that the receiver may force the sender to retransmit it. As either approach may be viewed as an attack (a sender spewing datagrams with spoofed addresses may clog its neighbors with traffic, and an application being forced to retransmit experiences a user-observable delay), this specification takes no position on that matter. The receiver MAY hold the datagram in a short queue to be operated on when the NA arrives, and it MAY discard it.

Given this model, there is a potential front-running attack on Stateless Address Autoconfiguration. When one system enters the Duplicate Address Detection phase, another system could see the address being verified and immediately send a message (perhaps an ICMP Echo Request sent as a link layer multicast) claiming the address. The systems that receive it would respond with a Neighbor Solicit, it would reply with a Neighbor Advertisement, and the original system would be denied the use of the address. As such, systems observing an address that they have no association for being verified using Duplicate Address Detection SHOULD NOT then grant it to another system.

As noted, in IPv6 networks hosts and routers usually have multiple addresses on any given interface. There is a potential attack in this as well: especially with privacy addressing, one could imagine a host using a new address for each TCP or SCTP session it opens, and one could imagine an attack that simulates this but simply fills its neighbor's tables with addresses. A host or router MAY (eg, is authorized to) limit the number of addresses for a neighbor that it will simultaneously hold, and the number of addresses considered reasonable is intentionally not specified. It SHOULD use memory allocated to that function in a Least Recently Used fashion, preserving recollection of addresses a neighbor is actually using and reusing table entries that appear to be unused.

As noted in Section 1.2 (Local Source Address Validation for IPv6), caveats that make this difficult include any system that might send a datagram from an address unknown in the local environment. These include, but are not limited to, routers and any middleware that behaves like a router, in that it forwards datagrams originated by other systems using their source addresses, and multi-LAN hosts. To simplify source address validation, this specification declares that a host SHOULD send any datagram it originates on an interface the source address is associated with. Routers and router-like middleware such as firewalls must be exlcuded from such analysis.

TOC

2.3. Validation of the addresses of neighboring systems by a switch

In this context, a "switch" refers to a system that switches messages for any lower layer protocol. Examples include Ethernet, ATM, Cable Modem, DSL, and so on.

As with IPv4, it is reasonable for a switch to simply observe the Neighbor Advertisements issued by a host or router and note that the host or router may be using them. There is a potential attack in this, however, in that a host could simply spew Neighbor Advertisements. For this reason, the protocol calls for a Neighbor Advertisement to be in response to a Neighbor Solicit. Therefore, a switch implementation that observes Neighbor Discovery or Secure Neighbor Discovery SHOULD remember addresses from Neighbor Advertisement only if it has seen a prior Neighbor Solicit.

A switch MAY observe DHCP assignments of addresses to hosts and drop host-originated datagrams using other source addresses. If it does, it SHOULD be prepared to accept multiple simultaneous assignments.

A switch or upstream router MAY also observe assignments of prefixes to downstream routers using the DHCP Prefix Assignment Options (Troan, O. and R. Droms, “IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6,” December 2003.) [RFC3633], and use that information to configure ingress prefix filtering.

As with host implementations, a switch MAY limit the number of addresses it will simultaneously store. If it does so, it SHOULD use memory allocated to that function in a Least Recently Used fashion, preserving recollection of addresses a neighbor is actually using and reusing table entries that appear to be unused.

3. Implementating Source Address Validation for remote systems

TOC

As mentioned, it is often in an administration's interest to protect itself from other administrations, which may have inadequate anti-spoofing measures in place. This is the original thrust of BCP 38 (Ferguson, P. and D. Senie, “Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing,” May 2000.) [RFC2827]. Unfortunately, one step removed from the source system, there is no anchor to verify that the address is exactly correct; rather, one can only verify that it is reasonable - it is within the prefix.

It has been argued that this is nonsensical, as anti-spoofing procedures impose additional router processing and only protect other networks. This is incorrect, however, for two reasons. Modern routers often implement filtering or Unicast Reverse Path Forwarding procedures in hardware, minimizing the processing burden, although the differentiated tables may consume memory. In any event, attacks perpetrated from a downstream network (obscured in some cases by address spoofing) attack both the administration's network and the administration's customers. Dealing with that problem is generally the first step in side-stepping an attack.

To implement, a router MAY be configured to discard traffic that routing believes is coming from an inappropriate direction. This does not depend on the router's choice of routes, however; it depends on the routing calculated by a router's neighbors. As such, if router A validly advertises to router B that it can route traffic to a prefix, the router B SHOULD accept traffic from that prefix via router A.

The determination of when a routing advertisement is valid is beyond the scope of this specification.

4. IANA Considerations

TOC

This memo adds no new IANA considerations.

Note to RFC Editor: This section will have served its purpose if it correctly tells IANA that no new assignments or registries are required, or if those assignments or registries are created during the RFC publication process. From the author's perspective, it may therefore be removed upon publication as an RFC at the RFC Editor's discretion.

5. Security Considerations

TOC

This note describes a set of security considerations for the IPv6 Internet, specifically related to attack management in IPv6 (Deering, S. and R. Hinden, “Internet Protocol, Version 6 (IPv6) Specification,” December 1998.) [RFC2460] and in the configuration and communication mechanisms of Stateless Address Autoconfiguration (Thomson, S., Narten, T., and T. Jinmei, “IPv6 Stateless Address Autoconfiguration,” September 2007.) [RFC4862], Neighbor Discovery (Narten, T., Nordmark, E., Simpson, W., and H. Soliman, “Neighbor Discovery for IP version 6 (IPv6),” September 2007.) [RFC4861], and SEcure Neighbor Discovery (Arkko, J., Kempf, J., Zill, B., and P. Nikander, “SEcure Neighbor Discovery (SEND),” March 2005.) [RFC3971]. It does not introduce new attacks, but it identifies some existing ones and recommends approaches to their mitigation.

TOC

6. Contributors

This document was developed in the SAVA context. Contributors include the author, Christian Vogt, Gang Ren, Hannes Tschofenig, Jari Arkko, Jianping Wu, Jun Bi, Ke Xu, and Pekka Savola.

7. References

TOC

7.1. Normative References

TOC

[RFC0791]	Postel, J., “Internet Protocol,” STD 5, RFC 791, September 1981 (TXT).
[RFC2460]	Deering, S. and R. Hinden, “Internet Protocol, Version 6 (IPv6) Specification,” RFC 2460, December 1998 (TXT, HTML, XML).
[RFC2827]	Ferguson, P. and D. Senie, “Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing,” BCP 38, RFC 2827, May 2000 (TXT).
[RFC3315]	Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, “Dynamic Host Configuration Protocol for IPv6 (DHCPv6),” RFC 3315, July 2003 (TXT).
[RFC3971]	Arkko, J., Kempf, J., Zill, B., and P. Nikander, “SEcure Neighbor Discovery (SEND),” RFC 3971, March 2005 (TXT).
[RFC4861]	Narten, T., Nordmark, E., Simpson, W., and H. Soliman, “Neighbor Discovery for IP version 6 (IPv6),” RFC 4861, September 2007 (TXT).
[RFC4862]	Thomson, S., Narten, T., and T. Jinmei, “IPv6 Stateless Address Autoconfiguration,” RFC 4862, September 2007 (TXT).
[RFC4941]	Narten, T., Draves, R., and S. Krishnan, “Privacy Extensions for Stateless Address Autoconfiguration in IPv6,” RFC 4941, September 2007 (TXT).

7.2. Informative References

TOC

[RFC2131]	Droms, R., “Dynamic Host Configuration Protocol,” RFC 2131, March 1997 (TXT, HTML, XML).
[RFC3633]	Troan, O. and R. Droms, “IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6,” RFC 3633, December 2003 (TXT).

[RFC3972]	Aura, T., “Cryptographically Generated Addresses (CGA),” RFC 3972, March 2005 (TXT).

Appendix A. Summary of recommendations

TOC

1. The NS/NA exchange on a response to a received datagram SHOULD happen before the received datagram is operated on.

2. The receiver MAY hold the triggering datagram in a short queue to be operated on when the NA arrives, and it MAY discard it.

3. Systems observing an address that they have no association for being verified using Duplicate Address Detection SHOULD NOT then grant it to another system.

4. A host or router MAY (eg, is authorized to) limit the number of addresses for a neighbor that it will simultaneously hold, and the number of addresses considered.

5. It SHOULD use memory allocated to that function in a Least Recently Used fashion, preserving recollection of addresses a neighbor is actually using and reusing table entries that appear to be unused.

6. To simplify source address validation, this specification declares that a host SHOULD send any datagram it originates on an interface the source address is associated with.

7. A switch that monitors Neighbor Discovery or Secure Neighbor Discovery SHOULD remember addresses from Neighbor Advertisement only if it has seen a prior Neighbor Solicit.

8. A switch MAY observe DHCP assignments of addresses to hosts and drop host-originated datagrams using other source addresses.

9. If it does, it SHOULD be prepared to accept multiple simultaneous assignments.

10. A switch or upstream router MAY also observe assignments of prefixes to downstream routers using the DHCP Prefix Assignment Options [RFC3633], and use that information to configure ingress prefix filtering.

11. A switch MAY limit the number of addresses it will simultaneously store.

12. If it does so, it SHOULD use memory allocated to that function in a Least Recently Used fashion, preserving recollection of addresses a neighbor is actually using and reusing table entries that appear to be unused.

13. A router MAY discard traffic that routing believes is coming from an inappropriate direction.

14. If router A validly advertises to router B that it can route traffic to a prefix, the router B SHOULD accept traffic from that prefix via router A.

Author's Address

TOC

Fred Baker
Cisco Systems
Santa Barbara, California 93117
USA
Phone: +1-408-526-4257
Email: fred@cisco.com

Full Copyright Statement

TOC

Intellectual Property