

Operational Security
Internet-Draft
Updates: 792, 4443 (if approved)
Intended status: Standards Track
Expires: April 10, 2013

F. Baker
G. Van de Velde
Cisco Systems
October 7, 2012

Passive IP Addresses
draft-baker-opsec-passive-ip-address-01

Abstract

This note suggests an approach to minimizing the attack surface of the network elements - routers, switches, and middleware - of a network.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 10, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
1.1. Requirements Language	3
1.2. Problem Statement	3
1.3. Examples of attacks	3
2. Proposal	4
2.1. Making the address useless	5
2.2. ICMP/ICMPv6 handling	6
2.3. Removing the address from routing	6
2.4. DNS and Reverse DNS	7
3. IANA Considerations	7
4. Security Considerations	7
4.1. Privacy Considerations	8
5. Acknowledgements	8
6. Change Log	8
7. References	8
7.1. Normative References	8
7.2. Informative References	8
Authors' Addresses	9

1. Introduction

This note suggests an approach to minimizing the attack surface of the network elements - routers, switches, and middleware - of a network.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

1.2. Problem Statement

The problem, at least in its first instance, is a side effect of diagnostics used in the Internet. Tools such as mtr, traceroute, and pingplotter operate by sending streams of packets to a remote address with varying hop limit values in IPv6 [RFC2460] or Time to Live in IPv4 [RFC0791], and receiving ICMP [RFC0792] or ICMPv6 [RFC4443] messages that indicate which interfaces the packet stream traversed in the forward direction. Path MTU [RFC1191] [RFC1981] discovery depends on ICMP/ICMPv6 Packet Too Big. Various ICMP/ICMPv6 "unreachable" messages respond when routing fails, which are intended to trigger applications to try other peer addresses [I-D.ietf-v6ops-happy-eyeballs], and so on. The IP addresses of these responders can be looked up in Reverse DNS [RFC1033][RFC1912] to build a name that indicates the operator, POP, and equipment in question, which is useful in identifying potential problems in the path.

Unfortunately, those addresses can also be used in another way. A motivated adversary can subject routers to TCP RST attacks, load-based DDOS, and other attacks.

An alternate way to reduce this potential attack vector is to not use addresses that are valid beyond the link it is attached towards. A solution describing considerations around this is given in [draft-ietf-opsec-lla-only] while passive IPv6 addresses will provide network path visibility without increasing large extend of vulnerability for the devices using down the traffic path.

1.3. Examples of attacks

To pick one example, attacks are being reported in which residential broadband customer's CPE Router is targeted with large volume SNMP GET Requests. The address of the router is not generally known; in IPv4, that may be a result of NAPT use, with the address being harvested from exchanges. It may be obtained from a traceroute to a

server behind the router, or it may be determined by analysis of SMTP envelopes.

Another example is attacks on BGP peering. BGP neighbors often peer between the loopback addresses of neighboring routers, to make the TCP session stable in the presence of link outages, but may peer using interface addresses. If a router is configured to use interface addresses in ICMP/ICMPv6 messages and to peer using those same addresses, the ICMP response exposes information that can be used in a RST attack on routing. It also facilitates any other kind of attack on the router, such as the previously noted SNMP attack (even if the router knows to refuse the message, it consumes CPU). If global addresses are not used - routers use link-local or private addresses - that makes it harder for an attacker to attack the router, but it means that traceroute and other uses are compromised, which is an attack on network forensics. If link-local addresses are used on the interfaces and ICMP is configured to use the loopback address, the router is again exposed to RST attacks.

2. Proposal

The simplest solution seems to be to enable the router to hide in plain sight - to use an address as the source address in ICMP and other messages that is identifiable using Reverse DNS (and therefore, through the name, useful for network diagnostics and communication between operators), but does not facilitate attacks.

The fundamental theory behind this proposal is the Principle of Least Privilege, which in this application is that an entity in the Internet must be able to access only the information and resources that are necessary for its legitimate purpose. In this case, it is reasonable, for various reasons, to enable a random user to identify the path his or her traffic is using or to identify a system in his path when reporting operational issues to an administration. It is not reasonable, or at least not required, that the user be able to specifically interact with any of those systems in the general case.

We propose that the source IP address in an ICMP/ICMPv6 message, or indeed any message sent to a host that has no inherent need to contact the specific system, be useful for Reverse DNS, but not for touching the system. Ideally, it is not routable to the system in the first place; The passive character of this type of address address comes to play if a packet with this address as destination address on a targetted device and is delivered to the interface, it is summarily dropped. Such an address is referred to as a "passive address", and if it comes from a specific prefix, the prefix is referred to as a "passive prefix". Addresses that are routable and not dropped on

receipt will, for the purposes of this specification, be called "active" IP addresses.

A passive address is semantically non-disguisable from any other type of address and has no requirement for any new type of address-family. Any IPv4 or IPv6 address can become a passive address by a configuration knob when specifying the interface IP address for the Interface or device.

2.1. Making the address useless

Every interface in the Internet has an address, with the exception of IPv4 unnumbered interfaces; even those have addresses that they use, which are the actual address of some other interface on the same system. Increasingly, this is in fact a list of addresses, some of which are IPv4 and some of which are IPv6.

We propose that any address allocated to an interface on infrastructure equipment be given two binary attributes:

UseInICMP: If the address has this attribute TRUE, the corresponding address may be used as the source address of ICMP or ICMPv6 messages and other messages sent to hosts that have no need to actually touch the system. It is otherwise FALSE.

Respond: If the address has this attribute TRUE, the device will process and respond to packets it receives that have this as a destination address; it is an active address. If the attribute is FALSE, the address is a passive address.

If UseInICMP is set TRUE on a Global Unicast Address or Unique Local Address, the address will be available for use in ICMP messages. If "Respond" is set TRUE, traffic sent to the address will be served in the usual way. This describes the present Internet usage. If Respond is set FALSE, traffic sent to the address will be summarily discarded, in effect presenting a "local firewall" blockage related to the address.

An address that has UseInICMP set FALSE will not be used as the source address of an ICMP message. That address will be undiscoverable via ICMP messages. If Respond is TRUE and the address becomes known by other means, such as DNS, traffic sent to the address will be served in the usual way. If Respond is set FALSE, traffic sent to the address will be summarily discarded, in effect presenting a "local firewall" blockage related to the address.

The scenario in view here is that

- o an address that is used to access the system would have UseInICMP FALSE (the address is not leaked in such messages) and Respond TRUE (messages sent to the address MAY be operated on by the system).
- o an address that is used in ICMP and similar messages would have UseInICMP TRUE (the address MAY be leaked in such messages) and Respond FALSE (messages sent to the address will be dropped on receipt).

2.2. ICMP/ICMPv6 handling

Per [RFC4443], an ICMP Response such as Time Exceeded or Parameter Problem is sent from "the" source address of the interface that detected the issue. This specification narrows that: it SHOULD use one of the source addresses that have the attribute UseInICMP set to TRUE. If no address has that attribute TRUE, it SHOULD NOT send the message.

2.3. Removing the address from routing

If the passive address is taken from any prefix that is not advertised in routing, it will be difficult for an adversary to route to the address, which simplifies the treatment of certain forms of attacks. It is not impossible; a system on the same LAN could send a crafted packet that would arrive anyway. However, especially in inter-domain routing, it is often quite reasonable to believe that addresses exist that need not be advertised to a neighboring network.

One example of such an address, in IPv6, might be a Unique Local IPv6 Unicast Address [RFC4193], or a global unicast address or prefix. There are obvious operational issues in the use of a global prefix; it is easy to accidentally advertise it. In an IPv4 network, the counterpart might be to use an [RFC1918] address, or to use another prefix that one chooses to not advertise.

Link Local addresses SHOULD NOT be used in this context; while they are obviously unroutable except on the local LAN, they are not useful in Reverse DNS.

One problem with this relates to Ingress Filtering [RFC2827]. If the prefix used for passive addresses is not advertised to the neighboring network and the neighboring network is using unicast reverse path filtering, it will filter these responses. For this reason, a network doing this SHOULD advise neighboring networks of passive prefixes for the purpose of inclusion in ingress filters.

2.4. DNS and Reverse DNS

[RFC1912] recommends that "For every IP address, there should be a matching PTR record in the in-addr.arpa domain." In IPv6, there is an important special case, in that link-local addresses are not reflected there, and are used in routing protocols for local communication among IPv6 routers. Like other addresses, passive IP addresses SHOULD have a corresponding Reverse DNS entry; these names help with traceroute and in fault diagnosis. While active addresses may be expected to have A or AAAA records in the administration's own DNS, there is little point for doing so for passive addresses, as they are unresponsive and very likely unreachable.

However, the names given to passive addresses SHOULD NOT be directly similar to the names given active IP addresses. For example, it may be useful to name the interfaces on a certain router so as to identify the router - "ethernet7.card3.router5.lax.example.com". If the correlation to the name of the loopback interface ("router5.lax.example.com") is obviously derivative, the security value is largely forfeit, although it might require human interaction. Such names should differ enough that they are not readily intuited, such as "rack12.lax.example.com".

3. IANA Considerations

This memo asks the IANA for no new parameters.

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.

4. Security Considerations

This entire note could be described as addressing a set of security considerations. It is not a complete solution to attacks on infrastructure - if loopback addresses, which are used for network management and other purposes are generally known, the infrastructure can still be attacked. However, it is an important reduction of the attack surface. It creates no attack surface that did not already exist.

4.1. Privacy Considerations

This proposal also introduces no new privacy issues.

5. Acknowledgements

This document grew from a conversation among the authors, John Brzozowski, and Thienpondt Hans. Merike Keao's review was very helpful.

6. Change Log

Initial Version: 1 March 2012

2th version: 7 October 2012

7. References

7.1. Normative References

- [RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981.
- [RFC0792] Postel, J., "Internet Control Message Protocol", STD 5, RFC 792, September 1981.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.
- [RFC4443] Conta, A., Deering, S., and M. Gupta, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", RFC 4443, March 2006.

7.2. Informative References

- [I-D.ietf-v6ops-happy-eyeballs]
Wing, D. and A. Yourtchenko, "Happy Eyeballs: Success with Dual-Stack Hosts", draft-ietf-v6ops-happy-eyeballs-07 (work in progress), December 2011.
- [RFC1033] Lottor, M., "Domain administrators operations guide", RFC 1033, November 1987.

- [RFC1191] Mogul, J. and S. Deering, "Path MTU discovery", RFC 1191, November 1990.
- [RFC1912] Barr, D., "Common DNS Operational and Configuration Errors", RFC 1912, February 1996.
- [RFC1918] Rekhter, Y., Moskowitz, R., Karrenberg, D., Groot, G., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, February 1996.
- [RFC1981] McCann, J., Deering, S., and J. Mogul, "Path MTU Discovery for IP version 6", RFC 1981, August 1996.
- [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.
- [RFC4193] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast Addresses", RFC 4193, October 2005.
- [draft-ietf-opsec-lla-only]
 , M., "Using Only Link-Local Addressing Inside an IPv6 Network", 20012.

Authors' Addresses

Fred Baker
Cisco Systems
Santa Barbara, California 93117
USA

Email: fred@cisco.com

Gunter Van de Velde
Cisco Systems
De Kleetlaan 6a
Diegem 1831
Belgium

Phone: +32 2704 5473
Email: gvandev@cisco.com

Operational Security Capabilities for
IP Network Infrastructure (opsec)
Internet-Draft
Intended status: BCP
Expires: April 26, 2013

F. Gont
SI6 Networks / UTN-FRH
W. Liu
Huawei Technologies
October 23, 2012

DHCPv6-Shield: Protecting Against Rogue DHCPv6 Servers
draft-gont-opsec-dhcpv6-shield-01

Abstract

This document specifies a mechanism for protecting hosts connected to a broadcast network against rogue DHCPv6 servers. The aforementioned mechanism is based on DHCPv6 packet-filtering at the layer-2 device on which the packets are received. The aforementioned mechanism has been widely deployed in IPv4 networks ('DHCP snooping'), and hence it is desirable that similar functionality be provided for IPv6 networks.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79. This document may not be modified, and derivative works of it may not be created, and it may not be published except as an Internet-Draft.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 26, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents

carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. DHCPv6-Shield Configuration	4
3. DHCPv6-Shield Implementation Advice	5
4. IANA Considerations	8
5. Security Considerations	9
6. Acknowledgements	10
7. References	11
7.1. Normative References	11
7.2. Informative References	11
Authors' Addresses	13

1. Introduction

This document specifies a mechanism for protecting hosts connected to a broadcast network against rogue DHCPv6 servers. This mechanism is analogous to the RA-Guard mechanism [RFC6104] [RFC6105] [I-D.ietf-v6ops-ra-guard-implementation] intended for protection against rogue Router Advertisement messages.

The basic concept behind DHCPv6-Shield is that a layer-2 device filters DHCPv6 messages meant to DHCPv6 clients, according to a number of different criteria. The most basic filtering criterion being that the aforementioned DHCPv6 messages are discarded by the layer-2 device unless they are received on a specified port of the layer-2 device.

Before the DHCPv6-Shield device is deployed, the administrator specifies the layer-2 port(s) on which DHCPv6 packets meant for DHCPv6 clients are allowed. Only those ports to which a DHCPv6 server is to be connected should be specified as such. Once deployed, the DHCPv6-Shield device inspects received packets, and allows (i.e. passes) DHCPv6 messages meant for DHCPv6 clients only if they are received on layer-2 ports that have been explicitly configured for such purpose.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

2. DHCPv6-Shield Configuration

Before being deployed for production, the DHCPv6-Shield device MUST be configured with respect to which layer-2 ports are allowed to send DHCPv6 packets to DHCPv6 clients. Only those layer-2 ports explicitly configured for such purpose will be allowed to send DHCPv6 packets to DHCPv6 clients.

3. DHCPv6-Shield Implementation Advice

The following filtering rules MUST be enforced as part of an DHCPv6-Shield implementation on those ports that are not allowed to send DHCPv6 packets to DHCPv6 clients:

1. DHCPv6-Shield MUST parse the IPv6 entire header chain present in the packet, to identify whether it is a DHCPv6 packet meant for a DHCPv6 client.

RATIONALE: [RFC6564] specifies a uniform format for IPv6 Extension Header, thus meaning that an IPv6 node can parse an IPv6 header chain even if it contains Extension Headers that are not currently supported by that node. Additionally, [I-D.ietf-6man-oversized-header-chain] requires that if a packet is fragmented, the first fragment contains the entire IPv6 header chain.

DHCPv6-Shield implementations MUST NOT enforce a limit on the number of bytes they can inspect (starting from the beginning of the IPv6 packet), since this could introduce false-positives: legitimate packets could be dropped simply because the DHCPv6-Shield device does not parse the entire IPv6 header chain present in the packet. An implementation that has such an implementation-specific limit MUST NOT claim compliance with this specification, and MUST pass the packet when such implementation-specific limit is reached.

2. When parsing the IPv6 header chain, if the packet is a first-fragment (i.e., a packet containing a Fragment Header with the Fragment Offset set to 0) and it fails to contain the entire IPv6 header chain (i.e., all the headers starting from the IPv6 header up to, and including, the upper-layer header), DHCPv6-Shield MUST drop the packet, and SHOULD log the packet drop event in an implementation-specific manner as a security fault.

RATIONALE: [I-D.ietf-6man-oversized-header-chain] specifies that the first-fragment (i.e., the fragment with the Fragment Offset set to 0) MUST contain the entire IPv6 header chain, and allows intermediate systems such as routers to drop those packets that fail to comply with this requirement.

NOTE: This rule should only be applied to IPv6 fragments with a Fragment Offset of 0 (non-first fragments can be safely passed, since they will never reassemble into a complete datagram if they are part of a DHCPv6 packet meant for a DHCPv6 client received on a port where such packets are not allowed).

3. When parsing the IPv6 header chain, if the packet is identified to be a DHCPv6 packet meant for a DHCPv6 client, DHCPv6-Shield MUST drop the packet, and SHOULD log the packet drop event in an implementation-specific manner as a security fault.
4. In all other cases, RA-Guard MUST pass the packet as usual.

NOTE: For the purpose of enforcing the DHCPv6-Shield filtering policy, an ESP header [RFC4303] should be considered to be an "upper-layer protocol" (that is, it should be considered the last header in the IPv6 header chain). This means that packets employing ESP would be passed by the DHCPv6-Shield device to the intended destination. If the destination host does not have a security association with the sender of the aforementioned IPv6 packet, the packet would be dropped. Otherwise, if the packet is considered valid by the IPsec implementation at the receiving host and encapsulates a DHCPv6 message, it is up to the receiving host what to do with such packet.

If a packet is dropped due to this filtering policy, then the packet drop event SHOULD be logged in an implementation-specific manner as a security fault. The logging mechanism SHOULD include a drop counter dedicated to DHCPv6-Shield packet drops.

In order to protect current end-node IPv6 implementations, Rule #2 has been defined as a default rule to drop packets that cannot be positively identified as not being DHCPv6 packets meant for DHCPv6 clients (because the packet is a fragment that fails to include the entire IPv6 header chain). This means that, at least in theory, DHCPv6-Shield could result in false-positive blocking of some legitimate (non DHCPv6-server) packets. However, as noted in [I-D.ietf-6man-oversized-header-chain], IPv6 packets that fail to include the entire IPv6 header chain are virtually impossible to police with state-less filters and firewalls, and hence are unlikely to survive in real networks. [I-D.ietf-6man-oversized-header-chain] requires that hosts employing fragmentation include the entire IPv6 header chain in the first fragment (the fragment with the Fragment Offset set to 0), thus eliminating the aforementioned false positives.

The aforementioned filtering rules implicitly handle the case of fragmented packets: if the DHCPv6-Shield device fails to identify the upper-layer protocol as a result of the use of fragmentation, the corresponding packets would be dropped.

Finally, we note that IPv6 implementations that allow overlapping fragments (i.e. that do not comply with [RFC5722]) might still be subject of DHCPv6-based attacks. However, a recent assessment of

IPv6 implementations [SI6-FRAG] with respect to their fragment reassembly policy seems to indicate that most current implementations comply with [RFC5722].

4. IANA Considerations

This document has no actions for IANA.

5. Security Considerations

The mechanism specified in this document can be used to mitigate DHCPv6-based attacks. Attack vectors based on other messages (such as ICMPv6 Router Advertisements) are out of the scope of this document.

As noted in Section 3, IPv6 implementations that allow overlapping fragments (i.e. that do not comply with [RFC5722]) might still be subject of DHCPv6-based attacks. However, most current implementations seem to comply with [RFC5722], and hence forbid IPv6 overlapping fragments.

We note that if an attacker sends a fragmented DHCPv6 packets on a port not allowed to send such packets, the first-fragment would be dropped, and the rest of the fragments would be passed. This means that the victim node would tie memory buffers for the aforementioned fragments, which would never reassemble into a complete datagram. If a large number of such packets were sent by an attacker, and the victim node failed to implement proper resource management for the fragment reassembly buffer, this could lead to a Denial of Service (DoS). However, this does not really introduce a new attack vector, since an attacker could always perform the same attack by sending forged fragmented datagram in which at least one of the fragments is missing. [CPNI-IPv6] discusses some resource management strategies that could be implemented for the fragment reassembly buffer.

6. Acknowledgements

This document is heavily based on the document [I-D.ietf-v6ops-ra-guard-implementation] authored by Fernando Gont. Thus, the author would like to thank Ran Atkinson, Karl Auer, Robert Downie, Washam Fan, David Farmer, Marc Heuse, Nick Hilliard, Ray Hunter, Joel Jaeggli, Simon Perreault, Arturo Servin, Gunter van de Velde, James Woodyatt, and Bjoern A. Zeeb, for providing valuable comments on [I-D.ietf-v6ops-ra-guard-implementation], on which this document is based.

7. References

7.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [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.
- [RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC 4303, December 2005.
- [RFC5722] Krishnan, S., "Handling of Overlapping IPv6 Fragments", RFC 5722, December 2009.
- [RFC6564] Krishnan, S., Woodyatt, J., Kline, E., Hoagland, J., and M. Bhatia, "A Uniform Format for IPv6 Extension Headers", RFC 6564, April 2012.

7.2. Informative References

- [RFC6104] Chown, T. and S. Venaas, "Rogue IPv6 Router Advertisement Problem Statement", RFC 6104, February 2011.
- [RFC6105] Levy-Abegnoli, E., Van de Velde, G., Popoviciu, C., and J. Mohacsi, "IPv6 Router Advertisement Guard", RFC 6105, February 2011.
- [I-D.ietf-6man-oversized-header-chain]
Gont, F. and V. Manral, "Security and Interoperability Implications of Oversized IPv6 Header Chains", draft-ietf-6man-oversized-header-chain-01 (work in progress), July 2012.
- [I-D.ietf-v6ops-ra-guard-implementation]
Gont, F., "Implementation Advice for IPv6 Router Advertisement Guard (RA-Guard)", draft-ietf-v6ops-ra-guard-implementation-04 (work in progress), May 2012.
- [SI6-FRAG]
SI6 Networks, "IPv6 NIDS evasion and improvements in IPv6 fragmentation/reassembly", 2012, <<http://blog.si6networks.com/2012/02/ipv6-nids-evasion-and-improvements-in.html>>.

[CPNI-IPv6]

Gont, F., "Security Assessment of the Internet Protocol
version 6 (IPv6)", UK Centre for the Protection of
National Infrastructure, (available on request).

Authors' Addresses

Fernando Gont
SI6 Networks / UTN-FRH
Evaristo Carriego 2644
Haedo, Provincia de Buenos Aires 1706
Argentina

Phone: +54 11 4650 8472
Email: fgont@si6networks.com
URI: <http://www.si6networks.com>

Will Liu
Huawei Technologies
Bantian, Longgang District
Shenzhen 518129
P.R. China

Email: liushucheng@huawei.com

Operational Security Capabilities for
IP Network Infrastructure (opsec)
Internet-Draft
Obsoletes: 5157 (if approved)
Intended status: Informational
Expires: April 26, 2013

F. Gont
Huawei Technologies
T. Chown
University of Southampton
October 23, 2012

Network Reconnaissance in IPv6 Networks
draft-gont-opsec-ipv6-host-scanning-02

Abstract

IPv6 offers a much larger address space than that of its IPv4 counterpart. The standard /64 IPv6 subnets can (in theory) accommodate approximately $1.844 * 10^{19}$ hosts, thus resulting in a much lower host density (#hosts/#addresses) than their IPv4 counterparts. As a result, it is widely assumed that it would take a tremendous effort to perform address scanning attacks against IPv6 networks, and therefore IPv6 address scanning attacks have long been considered unfeasible. This document analyzes how traditional address scanning techniques apply to IPv6 networks, and also explores a number of techniques that can be employed for IPv6 network reconnaissance.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79. This document may not be modified, and derivative works of it may not be created, and it may not be published except as an Internet-Draft.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 26, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Requirements for the Applicability of Network Reconnaissance Techniques	4
3. IPv6 Address scanning	5
3.1. Address configuration in IPv6	5
3.2. IPv6 address scanning of remote area networks	11
3.3. IPv6 address scanning of local area networks	11
3.4. Existing IPv6 address scanning tools	12
3.5. Mitigations	13
4. Leveraging the Domain Name System (DNS) for Network Reconnaissance	15
4.1. DNS Advertised Hosts	15
4.2. DNS Zone Transfers	15
4.3. Leveraging DNS reverse mappings for network reconnaissance	15
5. Public archives	17
6. Application Participation	18
7. Inspection of the IPv6 Neighbor Cache and Routing Table	19
8. Inspection of System Configuration and Log Files	20
9. Gleaning information from Routing Protocols	21
10. Security Considerations	22
11. Acknowledgements	23
12. References	24
12.1. Normative References	24
12.2. Informative References	24
Appendix A. Implementation of a full-fledged IPv6 address-scanning tool	27
A.1. Host-probing considerations	27
A.2. Implementation of an IPv6 local address-scanning tool	28
A.3. Implementation of a IPv6 remote address-scanning tool	29
Authors' Addresses	31

1. Introduction

The main driver for IPv6 deployment is its larger address space [CPNI-IPv6]. This larger address space not only allows for an increased number of connected devices, but also introduces a number of subtle changes in several aspects of the resulting networks. One of such changes is the reduced host density (Nr. of addresses/Nr. of hosts) of typical IPv6 subnetworks: with default IPv6 subnets of /64, each subnet comprises more than $1.844 * 10^{19}$ addresses; however, the actual number of nodes in each subnet is likely to remain similar to that of IPv4 subnetworks (at most a few hundred nodes per subnet). This lower host-density has led to the widely-established myth that IPv6 address-scanning attacks are unfeasible, since they would require a ridiculously long time (along with a tremendous amount of traffic) to be successfully performed.

This document analyzes the feasibility of "traditional" address-scanning attacks in IPv6 networks. Namely, it performs a thorough analysis of how IPv6 addresses are generated, and sheds some light on the real size of the search space for IPv6 address scanning attacks (e.g., "ping sweeps") thus dismantling the myth that such IPv6 address scanning attacks are unfeasible. Additionally, this document explores a number of other techniques that can be employed for IPv6 network reconnaissance.

On one hand, raising awareness about IPv6 network reconnaissance techniques may allow (in some cases) network and security administrators to prevent or detect such attempts. On the other hand, network reconnaissance is essential for the so-called "penetration tests" typically performed to assess the security of production networks. As a result, we believe the benefits of a thorough discussion of IPv6 network reconnaissance are two-fold.

Section 3 analyzes the feasibility of traditional address-scanning attacks (e.g. ping sweeps) in IPv6 networks, and explores a number of possible improvements to such techniques. [van-Dijk] describes a recently-disclosed technique for leveraging DNS reverse mappings for discovering IPv6 nodes. Finally, Appendix A describes how the analysis carried out throughout this document can be leveraged to produce an address-scanning tools (e.g. for penetration testing purposes).

2. Requirements for the Applicability of Network Reconnaissance Techniques

Throughout this document, a number of network reconnaissance techniques are discussed. Each of these techniques have different requirements on the side of the practitioner, with respect to whether they require local access to the target network, and whether they require login access to the system on which the technique is applied.

The following table tries to summarize the aforementioned requirements, and serve as a cross index to the corresponding sections.

Technique	Local access	Login access
Local address scans (Section 3.3)	Yes	No
Remote Address scans (Section 3.2)	No	No
DNS Advertised Hosts (Section 4.1)	No	No
DNS Zone Transfers (Section 4.2)	No	No
DNS reverse mappings (Section 4.3)	No	No
Public archives (Section 5)	No	No
Application Participation (Section 6)	No	No
Inspection of the IPv6 Neighbor Cache and Routing Table (Section 7)	No	Yes
Inspecting System Configuration and Log Files (Section 8)	No	Yes
Gleaning information from Routing Protocols (Section 9)	Yes	No

Table 1: Requirements for the Applicability of Network Reconnaissance Techniques

3. IPv6 Address scanning

This section discusses how traditional address scanning techniques (e.g. "ping sweeps") apply to IPv6 networks. Section 3.1 provides an essential analysis of how address configuration is performed in IPv6, identifying patterns in IPv6 addresses that can be leveraged to reduce the IPv6 address search space when performing IPv6 address scans. Appendix A discusses how the insights obtained in the previous sub-sections can be incorporated into a full-fledged IPv6 address scanning tool. Section 3.5 provides advice on how to mitigate IPv6 address scans.

3.1. Address configuration in IPv6

IPv6 incorporates two automatic address-configuration mechanisms: SLAAC (StateLess Address Auto-Configuration) [RFC4862] and DHCPv6 (Dynamic Host Configuration Protocol version 6) [RFC3315]. SLAAC is the mandatory mechanism for automatic address configuration, while DHCPv6 is optional - however, most current versions of general-purpose operating systems support both. In addition to automatic address configuration, hosts may employ manual configuration, in which all the necessary information is manually entered by the host or network administrator into configuration files at the host.

The following subsections describe each of the possible configuration mechanisms/approaches in more detail.

3.1.1. StateLess Address Auto-Configuration (SLAAC)

The basic idea behind SLAAC is that every host joining a network will send a multicasted solicitation requesting network configuration information, and local routers will respond to the request providing the necessary information. SLAAC employs two different ICMPv6 message types: ICMPv6 Router Solicitation and ICMPv6 Router Advertisement messages. Router Solicitation messages are employed by hosts to query local routers for configuration information, while Router Advertisement messages are employed by local routers to convey the requested information.

Router Advertisement messages convey a plethora of network configuration information, including the IPv6 prefix that should be used for configuring IPv6 addresses on the local network. For each local prefix learned from a Router Advertisement message, an IPv6 address is configured by appending a locally-generated Interface Identifier (IID) to the corresponding IPv6 prefix.

The following subsections describe currently-deployed policies for generating the IIDs used with SLAAC.

3.1.1.1. Interface-Identifiers embedding IEEE Identifiers

Many network technologies generate the 64-bit interface identifier based on the link-layer address of the corresponding network interface card. For example, in the case of Ethernet addresses, the IIDs are constructed as follows:

1. The "Universal" bit (bit 6, from left to right) of the address is set to 1
2. The word 0xffff is inserted between the OUI (Organizationally Unique Identifier) and the rest of the Ethernet address

For example, the MAC address 00:1b:38:83:88:3c would lead to the IID 021b:38ff:fe83:883c.

A number of considerations should be made about these identifiers. Firstly, as it should be obvious from the algorithm described above, two bytes (bytes 4-5) of the resulting address always have a fixed value (0xff, 0xfe), thus reducing the search space for the IID. Secondly, the first three bytes of these identifiers correspond to the OUI of the network interface card vendor. Since not all possible OUIs have been assigned, this further reduces the IID search space. Furthermore, of the assigned OUIs, many could be regarded as corresponding to legacy devices, and thus unlikely to be used for Internet-connected IPv6-enabled systems, yet further reducing the IID search space. Finally, in some scenarios it could be possible to infer the OUI in use by the target network devices, yet narrowing down the possible IIDs even more.

For example, an organization known for being provisioned by vendor X is likely to have most of the nodes in its organizational network with OUIs corresponding to vendor X.

These considerations mean that in some scenarios, the original IID search space of 64 bits may be effectively reduced to 2^{24} , or $n * 2^{24}$ (where "n" is the number of different OUIs assigned to the target vendor).

Another interesting factor arises from the use of virtualization technologies, since they generally employ automatically-generated MAC addresses, with very specific patterns. For example, all automatically-generated MAC addresses in VirtualBox virtual machines employ the OUI 08:00:27 [VBox2011]. This means that all SLAAC-produced addresses will have an IID of the form a00:27ff:feXX:XXXX, thus effectively reducing the IID search space from 64 bits to 24 bits.

VMWare ESX server provides yet a more interesting example. Automatically-generated MAC addresses have the following pattern [vmesx2011]:

1. The OUI is set to 00:05:59
2. The next 16-bits of the MAC address are set to the same value as the last 16 bits of the console operating system's primary IPv4 address
3. The final eight bits of the MAC address are set to a hash value based on the name of the virtual machine's configuration file.

This means that, assuming the console operating system's primary IPv4 address is known, the IID search space is reduced from 64 bits to 8 bits.

On the other hand, manually-configured MAC addresses in VMWare ESX server employ the OUI 00:50:56, with the low-order three bytes being in the range 0x000000-0x3fffff (to avoid conflicts with other VMware products). Therefore, even in the case of manually-configured MAC addresses, the IID search space is reduced from 64-bits to 22 bits.

3.1.1.2. Privacy Addresses

Privacy concerns [CPNI-IPv6] [Gont-DEEPSEC2011] regarding interface identifiers embedding IEEE identifiers led to the introduction of "Privacy Extensions for Stateless Address Auto-configuration in IPv6" [RFC4941], also known as "privacy addresses" or "temporary addresses". Essentially, "privacy addresses" produce random addresses by concatenating a random identifier to the auto-configuration IPv6 prefix advertised in a Router Advertisement.

In addition to their unpredictability, these addresses are typically short-lived, such that even if an attacker were to learn one of these addresses, they would be of use for a reduced period of time.

It is important to note that "privacy addresses" are generated in addition to traditional SLAAC addresses (i.e., based on IEEE identifiers): traditional SLAAC addresses are employed for incoming (i.e. server-like) communications, while "privacy addresses" are employed for outgoing (i.e., client-like) communications. This means that implementation/use of "privacy addresses" does not prevent an attacker from leveraging the predictability of traditional SLAAC addresses, since "privacy addresses" are generated in addition to (rather than in replacement of) the traditional SLAAC addresses derived from e.g. IEEE identifiers.

3.1.1.3. Stable and random Interface Identifiers

In order to mitigate the security implications arising from the predictable IPv6 addresses derived from IEEE identifiers, Microsoft Windows produced an alternative scheme for generating "stable addresses" (in replacement of the ones embedding IEEE identifiers). The aforementioned scheme is allegedly an implementation of RFC 4941 [RFC4941], but without regenerating the addresses over time. The resulting interface IDs are constant across system bootstraps, and also constant across networks.

Assuming no flaws in the aforementioned algorithm, this scheme would remove any patterns from the SLAAC addresses.

However, since the resulting interface IDs are constant across networks, these addresses may still be leveraged for host tracking purposes [I-D.ietf-6man-stable-privacy-addresses].

3.1.1.4. Stable Privacy-Enhanced Addresses

In response to the predictability issues discussed in Section 3.1.1.1 and the privacy issues discussed in , the IETF is currently standardizing (in [I-D.ietf-6man-stable-privacy-addresses]) a method for generating IPv6 Interface Identifiers to be used with IPv6 Stateless Address Autoconfiguration (SLAAC), such that addresses configured using this method are stable within each subnet, but the Interface Identifier changes when hosts move from one network to another. The aforementioned method is meant to be an alternative to generating Interface Identifiers based on IEEE identifiers, such that the benefits of stable addresses can be achieved without sacrificing the privacy of users.

Implementation of this method (in replacement of Interface Identifiers based on IEEE identifiers) would eliminate any patterns from the Interface ID.

3.1.2. Dynamic Host Configuration Protocol version 6 (DHCPv6)

DHCPv6 can be employed as a stateful address configuration mechanism, in which a server (the DHCPv6 server) leases IPv6 addresses to IPv6 hosts. As with the IPv4 counterpart, addresses are assigned according to a configuration-defined address range and policy, with some DHCPv6 servers assigned addresses sequentially, from a specific range. In such cases, addresses tend to be predictable.

For example, if the prefix 2001:db8::/64 is used for assigning addresses on the local network, the DHCPv6 server might (sequentially) assign addresses from the range 2001:db8::1 - 2001:

db8::100.

In most common scenarios, this means that the IID search space will be reduced from the original 64 bits, to 8 or 16 bits.

3.1.3. Manually-configured addresses

In some scenarios, node addresses may be manually configured. This is typically the case for IPv6 addresses assigned to routers, since routers do not employ automatic address configuration.

While network administrators are mostly free to select the IID from any value in the range 1 - 264 range, for the sake of simplicity (i.e., ease of remembering) they tend to select addresses with one of the following patterns:

- o "low-byte" addresses: in which all bytes of the IID (except the lowest one) are set to 0.
- o IPv4-based addresses: in which the IID encodes the IPv4-address of the network interface (as in 2001:db8::192.168.1.1)
- o wordy addresses: which encode words (as in 2001:db8::dead:beef)

Clearly, the first two patterns reduce the search space from the original 64 bits to roughly 8 bits (assuming the IPv4 address range is known for the case of "IPv4-based" addresses). On the other hand, the search space for IPv6 wordy-addresses is probably larger and more complex, but still greatly reduced when compared to the original 64-bit search space.

3.1.4. IPv6 addresses corresponding to transition/co-existence technologies

Some transition/co-existence technologies might be leveraged to reduce the target search space of remote address-scanning attacks, since they specify how the corresponding IPv6 address must be generated. For example, in the case of Teredo [RFC4380], the 64-bit interface identifier is generated from the IPv4 address observed at a Teredo server along with a UDP port number.

3.1.5. IPv6 address assignment in real-world network scenarios

Table 2 and Table 3 provide a rough summary of the results obtained by [Malone2008] for IPv6 clients and IPv6 routers, respectively. These results are provided mainly for completeness-sake, since they are the most comprehensive address-measurement results that have so far been made publicly available.

We note, however, that evolution of IPv6 implementations, changes in the IPv6 address selection policy, etc., might limit (or even obsolete) the validity of these results.

Address type	Percentage
SLAAC	50%
IPv4-based	20%
Teredo	10%
Low-byte	8%
Privacy	6%
Wordy	<1%
Other	<1%

Table 2: Measured client addresses

Address type	Percentage
Low-byte	70%
IPv4-based	5%
SLAAC	1%
Wordy	<1%
Privacy	<1%
Teredo	<1%
Other	<1%

Table 3: Measured router addresses

It should be clear from these measurements that a very high percentage of the client addresses follow very specific patterns.

3.2. IPv6 address scanning of remote area networks

While in IPv4 networks attackers have been able to get away with "brute force" scanning attacks (thanks to the reduced search space), successfully performing a brute-force scan of an entire /64 network would be infeasible. As a result, it is expected that attackers will leverage the IPv6 address patterns discussed in Section 3.1 to reduce the IPv6 address search space.

IPv6 address scanning of remote area networks should consider an additional factor not present for the IPv4 case: since the typical IPv6 subnet is a /64, scanning an entire /64 could, in theory, lead to the creation of 2^{64} entries in the Neighbor Cache of the last-hop router. Unfortunately, a number of IPv6 implementations have been found to be unable to properly handle large number of entries in the Neighbor Cache, and hence these address-scan attacks may have the side effect of resulting in a Denial of Service (DoS) attack [CPNI-IPv6] [I-D.ietf-v6ops-v6nd-problems].

3.3. IPv6 address scanning of local area networks

IPv6 address scanning in Local Area Networks could be considered, to some extent, a completely different problem than that of scanning a remote IPv6 network. The main difference is that use of link-local multicast addresses can relieve the attacker of searching for unicast addresses in a large IPv6 address space.

Obviously, a number of other network reconnaissance vectors (such as network snooping, leveraging Neighbor Discovery traffic, etc.) are available when scanning a local network. However, this section focuses only on address-scanning attacks (a la "ping sweep").

An attacker can simply send probe packets to the all-nodes link-local multicast address (ff02::1), such that responses are elicited from all local nodes.

Since Windows systems (Vista, 7, etc.) do not respond to ICMPv6 Echo Request messages sent to multicast addresses, IPv6 address-scanning tools typically employ a number of additional probe packets to elicit responses from all the local nodes. For example, unrecognized IPv6 options of type 10xxxxxx elicit ICMPv6 Parameter Problem, code 2, error messages.

Many address-scanning tools discover only IPv6 link-local addresses (rather than e.g. the global addresses of the target systems): since the probe packets are typically sent with the attacker's IPv6 link-local address, the "victim" nodes send the response packets using the

IPv6 link-local address of the corresponding network interface (as specified by the IPv6 address selection rules [RFC3484]). However, sending multiple probe packets, with each packet employing addresses from different prefixes, typically helps to overcome this limitation.

This technique is employed by the scan6 tool of the IPv6 Toolkit package [IPv6-Toolkit].

3.4. Existing IPv6 address scanning tools

3.4.1. Remote IPv6 network scanners

IPv4 address scanning tools have traditionally carried out their task for probing an entire address range (usually the entire range of a target subnetwork). One might argue that the reason for which we have been able to get away with such somewhat "rudimentary" techniques is that the scale of the "problem" is so small in the IPv4 world, that a "brute-force" attack is "good enough". However, the scale of the "address scanning" problem is so large in IPv6, that attackers must be very creative to be "good enough".

Simply sweeping an entire /64 IPv6 subnet would just not be feasible. For instance, that is one of the reasons for which address scanning tools such as nmap [nmap2012] do not even support sweeping an IPv6 address range.

The nmap(1) manual page states "IPv6 addresses can only be specified by their fully qualified IPv6 address or hostname. CIDR and octet ranges aren't supported for IPv6 because they are rarely useful.

On the other hand, the alive6 tool from [THC-IPv6] supports sweeping address ranges, thus being able to leverage some patterns found in IPv6 addresses, such as the incremental addresses resulting from some DHCPv6 setups.

The most "advanced" IPv6 scanning technique that has been found in the wild is that reported in [Ybema2010], in which the attacker seemed to be scanning specific IPv6 addresses based on specific patterns. However, the aforementioned attempt probably still falls into the category of "rudimentary".

Clearly, a limitation of most currently-available tools is that they lack of an "heuristics engine" that can help reduce the search space, such that the problem of IPv6 address scanning becomes tractable. However, we expect that this situation will change in the short term.

3.4.2. Local IPv6 network scanners

There are a variety of publicly-available local IPv6 network scanners:

Current versions of nmap [nmap2012] implement this functionality

THC's IPv6 Attack Toolkit [THC-IPV6] includes a tool that implements this functionality

SI6 Network's IPv6 Toolkit [IPv6-Toolkit] includes a tool (scan6) that implements this functionality

3.5. Mitigations

IPv6 address-scanning attacks can be mitigated in a number of ways. A non-exhaustive list of the possible mitigations includes:

- o Employing stable privacy-enhanced addresses [I-D.ietf-6man-stable-privacy-addresses] in replacement of addresses based on IEEE identifiers, such that any address patterns are eliminated.
- o Employing Intrusion Prevention Systems (IPS) at the perimeter, such that address scanning attacks can be mitigated.
- o If virtual machines are employed, and "resistance" to address scanning attacks is deemed as desirable, manually-configured MAC addresses can be employed, such that even if the virtual machines employ IEEE-derived IIDs, they are generated from non-predictable MAC addresses.

It should be noted that some of the aforementioned mitigations are operational, while others depend on the availability of specific features (such as [I-D.ietf-6man-stable-privacy-addresses] on the corresponding nodes.

Additionally, while some resistance to address scanning attacks is generally desirable (particularly when lightweight mitigations are available), there are scenarios in which mitigation of some address-scanning vectors is unlikely to be a high-priority (if at all possible).

Two of the techniques discussed in this document for local address-scanning attacks are those that employ multicasted ICMPv6 Echo Requests and multicasted IPv6 packets containing unsupported options of type 10xxxxxx. These two vectors could be easily mitigated by configuring nodes to not respond to multicasted ICMPv6 Echo Request

(default on Windows systems), and by updating the IPv6 specifications (and/or possibly configuring local nodes) such that multicasted packets never elicit ICMPv6 error messages (even if they contain unsupported options of type 10xxxxxx).

[I-D.gont-6man-ipv6-smurf-amplifier] proposes such update to the IPv6 specifications.

In any case, when it comes to local networks, there are a variety of network reconnaissance vectors. Therefore, even if address-scanning vectors are mitigated, an attacker could still rely on e.g. protocols employed for the so-called "opportunistic networking" (such as mDNS), or eventually on network snooping, for the purpose of network reconnaissance.

4. Leveraging the Domain Name System (DNS) for Network Reconnaissance

4.1. DNS Advertised Hosts

Any systems that are "published" in the DNS, e.g. MX mail relays, or web servers, will remain open to probing from the very fact that their IPv6 addresses are publicly available. It is worth noting that where the addresses used at a site follow specific patterns, publishing just one address may lead to a threat upon the other hosts.

Additionally, we note that publication of IPv6 addresses in the DNS should not discourage the elimination of IPv6 address patterns: if any address patterns are eliminated from addresses published in the DNS, an attacker may have to rely on performing dictionary-based DNS lookups in order to find all systems in a target network (which is generally less reliable and more time/traffic consuming than mapping nodes with predictable IPv6 addresses).

4.2. DNS Zone Transfers

A DNS zone transfer can readily provide information about potential attack targets. Restricting zone transfers is thus probably more important for IPv6, even if it is already good practice to restrict them in the IPv4 world.

4.3. Leveraging DNS reverse mappings for network reconnaissance

An interesting technique that employs DNS reverse mappings for network reconnaissance has been recently disclosed [van-Dijk]. Essentially, the attacker walks through the "ip6.arpa" zone looking up PTR records, in the hopes of learning the IPv6 addresses of hosts in a given target network (assuming that the reverse mappings have been configured, of course). What is most interesting about this technique is that it can greatly reduce the IPv6 address search space.

Basically, an attacker would walk the ip6.arpa zone corresponding to a target network (e.g. "0.8.0.0.8.b.d.0.1.0.0.2.ip6.arpa." for "2001:db8:80:/32"), issuing queries for PTR records corresponding to the domain names "0.0.8.0.0.8.b.d.0.1.0.0.2.ip6.arpa.", "1.0.8.0.0.8.b.d.0.1.0.0.2.ip6.arpa.", etc. If, say, there were PTR records for any hosts "starting" with the domain name "0.0.8.0.0.8.b.d.0.1.0.0.2.ip6.arpa." (e.g., the ip6.arpa domain name corresponding to the IPv6 address 2001:db8:80::1), the response would contain an RCODE of 0 (no error). Otherwise, the response would contain an RCODE of 4 (NXDOMAIN). As noted in [van-Dijk], this technique allows for a tremendous reduction in the "IPv6 address"

search space.

5. Public archives

Public mailing-list archives or Usenet news messages archives may prove a useful channel for an attacker, since hostnames and/or IPv6 addresses could be easily obtained by inspection of the (many) "Received from:" or other header lines in the archived email or Usenet news messages.

6. Application Participation

Peer-to-peer applications often include some centralised server which coordinates the transfer of data between peers. For example, BitTorrent builds swarms of nodes that exchange chunks of files, with a tracker passing information about peers with available chunks of data between the peers. Such applications may offer an attacker a source of peer addresses to probe.

7. Inspection of the IPv6 Neighbor Cache and Routing Table

Information about other systems connected to the local network might be readily available from the Neighbor Cache [RFC4861] and/or the routing table of any system connected to such network.

While the requirement of having "login" access to a system in the target network may limit the applicability of this technique, there are a number of scenarios in which this technique might be of use. For example, security audit tools might be provided with the necessary credentials such that the Neighbor Cache and the routing table of all systems for which the tool has "login" access can be automatically gleaned. On the other hand, IPv6 worms [V6-WORMS] could leverage this technique for the purpose of spreading on the local network, since they will typically have access to the Neighbor Cache and routing table of an infected system.

8. Inspection of System Configuration and Log Files

Nodes are generally configured with the addresses of other important local computers, such as email servers, local file servers, web proxy servers, recursive DNS servers, etc. The `/etc/hosts` file in UNIX, SSH `known_hosts` files, or the Microsoft Windows registry are just some examples of places where interesting information about such systems might be found.

Additionally, system log files (including web server logs, etc.) may also prove a useful channel for an attacker.

While the required credentials to access the aforementioned configuration and log files may limit the applicability of this technique, there are a number of scenarios in which this technique might be of use. For example, security audit tools might be provided with the necessary credentials such that these files can be automatically accessed. On the other hand, IPv6 worms could leverage this technique for the purpose of spreading on the local network, since they will typically have access to these files on an infected system [V6-WORMS].

9. Gleaning information from Routing Protocols

Some organizational IPv6 networks employ routing protocols to dynamically maintain routing information. In such an environment, a local attacker could become a passive listener of the routing protocol, to determine other valid subnets within that organization [V6-WORMS].

10. Security Considerations

This document explores the topic of Network Reconnaissance in IPv6 networks. It analyzes the feasibility of address-scan attacks in IPv6 networks, and showing that the search space for such attacks is typically much smaller than the one traditionally assumed (64 bits). Additionally, it explores a plethora of other network reconnaissance techniques, ranging from inspecting the IPv6 Network Cache of an attacker-controlled system, to gleaning information about IPv6 addresses from public mailing-list archives or Peer-To-Peer (P2P) protocols.

We expect traditional address-scanning attacks to become more and more elaborated (i.e., less "brute force"), and other network reconnaissance techniques to be actively explored, as global deployment of IPv6 increases and, more specifically, as more IPv6-only devices are deployed.

11. Acknowledgements

The author would like to thank (in alphabetical order) Marc Heuse, Ray Hunter, Libor Polcak, Jan Schaumann, and Arturo Servin, for providing valuable comments on earlier versions of this document.

Part of the contents of this document are based on the results of the project "Security Assessment of the Internet Protocol version 6 (IPv6)" [CPNI-IPv6], carried out by Fernando Gont on behalf of the UK Centre for the Protection of National Infrastructure (CPNI). Fernando Gont would like to thank the UK CPNI for their continued support.

12. References

12.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.
- [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.
- [RFC3484] Draves, R., "Default Address Selection for Internet Protocol version 6 (IPv6)", RFC 3484, February 2003.
- [RFC4380] Huitema, C., "Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs)", RFC 4380, February 2006.
- [RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, September 2007.
- [RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.
- [RFC4941] Narten, T., Draves, R., and S. Krishnan, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", RFC 4941, September 2007.
- [I-D.ietf-6man-stable-privacy-addresses] Gont, F., "A method for Generating Stable Privacy-Enhanced Addresses with IPv6 Stateless Address Autoconfiguration (SLAAC)", draft-ietf-6man-stable-privacy-addresses-01 (work in progress), October 2012.

12.2. Informative References

- [I-D.ietf-v6ops-v6nd-problems] Gashinsky, I., Jaeggli, J., and W. Kumari, "Operational Neighbor Discovery Problems", draft-ietf-v6ops-v6nd-problems-05 (work in progress), March 2012.
- [I-D.gont-6man-ipv6-smurf-amplifier] Gont, F., "Security Implications of IPv6 options of Type

10xxxxxx", draft-gont-6man-ipv6-smurf-amplifier-00 (work in progress), December 2011.

[RFC5157] Chown, T., "IPv6 Implications for Network Scanning", RFC 5157, March 2008.

[CPNI-IPv6] Gont, F., "Security Assessment of the Internet Protocol version 6 (IPv6)", UK Centre for the Protection of National Infrastructure, (available on request).

[V6-WORMS] Bellovin, S., Cheswick, B., and A. Keromytis, "Worm propagation strategies in an IPv6 Internet", ;login:, pages 70-76, February 2006, <<https://www.cs.columbia.edu/~smb/papers/v6worms.pdf>>.

[Malone2008] Malone, D., "Observations of IPv6 Addresses", Passive and Active Measurement Conference (PAM 2008, LNCS 4979), April 2008, <<http://www.maths.tcd.ie/~dwmalone/p/addr-pam08.pdf>>.

[nmap2012] Fyodor, "nmap - Network exploration tool and security / port scanner", 2012, <<http://insecure.org>>.

[VBox2011] VirtualBox, "Oracle VM VirtualBox User Manual, version 4.1.2", August 2011, <<http://www.virtualbox.org>>.

[vmesx2011] vmware, "Setting a static MAC address for a virtual NIC", vmware Knowledge Base, August 2011, <http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=219>.

[Ybema2010] Ybema, I., "just seen my first IPv6 network abuse scan, is this the start for more?", Post to the NANOG mailing-list, 2010, <<http://mailman.nanog.org/pipermail/nanog/2010-September/025049.html>>.

[Gont-DEEPSEC2011] Gont, "Results of a Security Assessment of the Internet Protocol version 6 (IPv6)", DEEPSEC 2011 Conference, Vienna, Austria, November 2011, <<http://www.sifnetworks.com/presentations/deepsec2011/>>.

fgont-deepsec2011-ipv6-security.pdf>.

[THC-IPV6]

"THC-IPV6", <<http://www.thc.org/thc-ipv6/>>.

[IPv6-Toolkit]

"IPv6 Toolkit",
<<http://www.si6networks.com/research/tools.html>>.

[van-Dijk]

van Dijk, P., "Finding v6 hosts by efficiently mapping
ip6.arpa", <[http://7bits.nl/blog/2012/03/26/
finding-v6-hosts-by-efficiently-mapping-ip6-arpa](http://7bits.nl/blog/2012/03/26/finding-v6-hosts-by-efficiently-mapping-ip6-arpa)>.

Appendix A. Implementation of a full-fledged IPv6 address-scanning tool

This section describes the implementation of a full-fledged IPv6 address scanning tool. Appendix A.1 discusses the selection of host probes. Appendix A.2 describes the implementation of an IPv6 address scanner for local area networks. Appendix A.3 outlines ongoing work on the implementation of a general (i.e., non-local) IPv6 host scanner.

A.1. Host-probing considerations

A number of factors should be considered when selecting the probe types and the probing-rate for an IPv6 address scanning tool.

Firstly, some hosts (or border firewalls) might be configured to block or rate-limit some specific packet types. For example, it is usual for host and router implementations to rate-limit ICMPv6 error traffic. Additionally, some firewalls might be configured to block or rate-limit incoming ICMPv6 echo request packets.

As noted earlier in this document, Windows systems simply do not respond to ICMPv6 echo requests sent to multicast IPv6 addresses.

Among the possible probe types are:

- o TCP segments meant to elicit SYN/ACK or RST segments,
- o UDP segments meant to elicit a UDP application response or an ICMPv6 Port Unreachable, an IPv6 packet containing any suitable payload and an unrecognized extension header (such that a ICMPv6 Parameter Problem error message is elicited), or,
- o an IPv6 packet containing any suitable payload and an unrecognized option of type 10xxxxxx (such that a ICMPv6 Parameter Problem error message is elicited)

Selecting an appropriate probe packet might help conceal the ongoing attack, but may also be actually necessary if host or network configuration causes certain probe packets to be dropped. In some cases, it might be desirable to insert some IPv6 extension headers before the actual payload, such that some filtering policies can be circumvented.

Another factor to consider is the host-probing rate. Clearly, the higher the rate, the smaller the amount of time required to perform the attack. However, the probing-rate should not be too high, or else:

1. the attack might cause network congestion, thus resulting in packet loss
2. the attack might hit rate-limiting, thus resulting in packet loss
3. the attack might reveal underlying problems in the Neighbor Discovery implementation, thus leading to packet loss and possibly even Denial of Service

Packet-loss is undesirable, since it would mean that an "alive" node might remain undetected as a result of a lost probe or response. Such losses could be the result of congestion (in case the attacker is scanning a target network at a rate higher than the target network can handle), or may be the result of rate-limiting as it would be typically the case if ICMPv6 is employed for the probe packets. Finally, as discussed in [CPNI-IPv6] and [I-D.ietf-v6ops-v6nd-problems], some IPv6 router implementations have been found to be unable to perform decent resource management when faced with Neighbor Discovery traffic involving a large number of local nodes. This essentially means that regardless of the type of probe packets, a address scanning attack might result in a Denial of Service (DoS) of the target network, with the same (or worse) effects as that of network congestion or rate-limiting.

The specific rates at which each of these issues may come into play vary from one scenario to another, and depend on the type of deployed routers/firewalls, configuration parameters, etc.

A.2. Implementation of an IPv6 local address-scanning tool

scan6 [IPv6-Toolkit] is prototype IPv6 local address scanning tool, which has proven to be effective and efficient for the discovery of IPv6 hosts on a local network.

The scan6 tool operates (roughly) as follows:

1. The tool learns the local prefixes used for auto-configuration, and generates/configures one address for each local prefix (in addition to a link-local address)
2. An ICMPv6 Echo Request message destined to the all-nodes on-link multicast address (ff02::1) is sent with each of the addresses "configured" in the previous step. Because of the different Source Addresses, each probe causes the victim nodes to use different Source Addresses for the response packets (this allows the tool to learn virtually all the addresses in use in the local network segment).

3. The same procedure of the previous bullet is performed, but this time with ICMPv6 packets that contain an unrecognized option of type 10xxxxxx, such that ICMPv6 Parameter Problem error messages are elicited. This allows the tool to discover e.g. Windows nodes, which otherwise do not respond to multicasted ICMPv6 Echo Request messages.
4. Each time a new "alive" address is discovered, the corresponding Interface-ID is combined with all the local prefixes, and the resulting addresses are probed (with unicasted packets). This can help to discover other addresses in use on the local network segment, since the same Interface ID is typically used with all the available prefixes for the local network.

The aforementioned scheme can fail to discover some addresses for some implementation. For example, Mac OS X employs IPv6 addresses embedding IEEE-identifiers (rather than "privacy addresses") when responding to packets destined to a link-local multicast address, sourced from an on-link prefix.

A.3. Implementation of a IPv6 remote address-scanning tool

An IPv6 remote address scanning tool, could be implemented with the following features:

- o The tool can be instructed to scan devices manufactured by a specific vendor, such that only addresses resulting for the corresponding OUIs are tried
- o The tool can be instructed to discover virtual machines, such that a given IPv6 prefix is only scanned for the address patterns resulting from virtual machines (as discussed earlier in this document)
- o The tool can be instructed to scan for low-byte or DHCPv6-like addresses
- o The tool can be instructed to scan for wordy-addresses, in which case the tool selects addresses based on a local dictionary
- o The tool can be specified an IPv4 address range in use at the target network, such that only IPv4-based IPv6 addresses are scanned.

In brute force mode, the tool can, at the very least:

- o Skip addresses resulting from unassigned OUIs

- o Skip addresses resulting from OUIs deemed as "legacy"

Authors' Addresses

Fernando Gont
Huawei Technologies
Evaristo Carriego 2644
Haedo, Provincia de Buenos Aires 1706
Argentina

Phone: +54 11 4650 8472
Email: fgont@si6networks.com
URI: <http://www.si6networks.com>

Tim Chown
University of Southampton
Highfield
Southampton, Hampshire SO17 1BJ
United Kingdom

Email: tjc@ecs.soton.ac.uk

Operational Security Capabilities for
IP Network Infrastructure (opsec)
Internet-Draft
Intended status: Informational
Expires: April 18, 2013

F. Gont
Huawei Technologies
October 15, 2012

Virtual Private Network (VPN) traffic leakages in dual-stack hosts/
networks
draft-gont-opsec-vpn-leakages-00

Abstract

The subtle way in which the IPv6 and IPv4 protocols co-exist in typical networks, together with the lack of proper IPv6 support in popular Virtual Private Network (VPN) products, may inadvertently result in VPN traffic leaks. That is, traffic meant to be transferred over a VPN connection may leak out of such connection and be transferred in the clear on the local network. This document discusses some scenarios in which such VPN leakages may occur, either as a side effect of enabling IPv6 on a local network, or as a result of a deliberate attack from a local attacker. Additionally, it discusses possible mitigations for the aforementioned issue.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79. This document may not be modified, and derivative works of it may not be created, and it may not be published except as an Internet-Draft.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 18, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. IPv4 and IPv6 co-existence	4
3. Virtual Private Networks in IPv4/IPv6 dual-stack hosts/networks	5
4. VPN traffic-leakages in legitimate scenarios	6
5. VPN traffic-leakage attacks	7
6. Mitigations to VPN traffic-leakage vulnerabilities	8
7. IANA Considerations	9
8. Security Considerations	10
9. Acknowledgements	11
10. References	12
10.1. Normative References	12
10.2. Informative References	12
Author's Address	14

1. Introduction

It is a very common practice for employees working at remote locations to establish a VPN connection with their office or home office. This is typically done to gain access to some resources only available within the company's network, but also to secure the host's traffic against attackers that might be connected to the same remote location. In some scenarios, it is even assumed that employing a VPN connection makes the use of insecure protocols (e.g. that transfer sensitive information in the clear) acceptable, as the VPN provides security services (such as confidentiality) for all communications made over the VPN.

Many VPN products that are typically employed for the aforementioned VPN connections only support the IPv4 protocol: that is, they perform the necessary actions such that IPv4 traffic is sent over the VPN connection, but they do nothing to secure IPv6 traffic originated from (or being received at) the host employing the VPN client. However, the hosts themselves are typically dual-stacked: they support (and enable by default) both IPv4 and IPv6 (even if such IPv6 connectivity is simply "dormant" when they connect to IPv4-only networks). When the IPv6 connectivity of such hosts is enabled, they may end up employing an IPv6-unaware VPN client in a dual-stack network. This may have "unexpected" consequences, as explained below.

The subtle way in which the IPv4 and IPv6 protocols interact and co-exist in dual-stacked networks might, either inadvertently or as a result of a deliberate attack, result in VPN traffic leakages -- that is, traffic meant to be transferred over a VPN connection could leak out of the VPN connection and be transmitted in the clear on the local network, without employing the VPN services at all.

Section 2 provides some background about IPv6 and IPv4 co-existence, summarizing how IPv4 and IPv4 interact on a typical dual-stacked network. Section 3 describes the underlying problem that leads to the aforementioned VPN traffic leakages. Section 4 describes legitimate scenarios in which such traffic leakages might occur, while Section 5 describes how VPN traffic leakages can be triggered by deliberate attacks.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

2. IPv4 and IPv6 co-existence

The co-existence of the IPv4 and IPv6 protocols has a number of interesting and subtle aspects that may have "surprising" consequences. While IPv6 is not backwards-compatible with IPv4, the two protocols are "glued" together by the Domain Name System (DNS).

For example, consider a site (say, `www.example.com`) that has both IPv4 and IPv6 support. The corresponding domain name (`www.example.com`, in our case) will contain both A and AAAA DNS resource records (RRs). Each A record will contain one IPv4 address, while each AAAA record will contain one IPv6 address -- and there might be more than one instance of each of these record types. Thus, when a dual-stacked client application means to communicate with the aforementioned site, it can request both A and AAAA records, and use any of the available addresses. The preferred address family (IPv4 or IPv6) and the specific address that will be used (assuming more than one address of each family is available) varies from one protocol implementation to another, with many host implementations preferring IPv6 addresses over IPv4 addresses.

[RFC6724] specifies an algorithm for selecting a destination address from a list of IPv6 and IPv4 addresses. [RFC6555] discusses the challenge of selecting the most appropriate destination address, along with a proposed implementation approach that mitigates connection-establishment delays.

This "co-existence" between IPv6 and IPv4 means that, when a dual-stacked client means to communicate with some other system, the availability of A and AAAA DNS resource records will typically affect which protocol is employed to communicate with that system.

3. Virtual Private Networks in IPv4/IPv6 dual-stack hosts/networks

Many Virtual Private Network (VPN) implementations do not support the IPv6 protocol -- or, what is worse, they completely ignore IPv6. This typically means that, when establishing a VPN connection, the VPN software takes care of the IPv4 connectivity by, e.g. inserting an IPv4 default route that causes all IPv4 traffic to be sent over the VPN connection (as opposed to sending the traffic in the clear, employing the local router). However, if IPv6 is not supported (or completely ignored), any packets destined to an IPv6 address will be sent in the clear using the local IPv6 router. That is, the VPN software will do nothing about the IPv6 traffic.

The underlying problem here is that while IPv4 and IPv6 are two different protocols incompatible with each other, the two protocols are glued together by the Domain Name System. Therefore, for dual-stacked systems, it is not possible to secure secure the communication with another system without securing both protocols (IPv6 and IPv4).

4. VPN traffic-leakages in legitimate scenarios

Consider a dual-stacked host that employs IPv4-only VPN software to establish a VPN connection with a VPN server, and that the host now attaches to a dual-stacked network (that provides both IPv6 and IPv4 connectivity). If some application on the client needs to communicate with a dual-stacked destination, the client will typically query both A and AAAA DNS resource records. Since the host will have both IPv4 and IPv6 connectivity, and the intended destination will have both A and AAAA DNS resource records, one of the possible outcomes is that the host will employ IPv6 to communicate with the aforementioned system. Since the VPN software does not support IPv6, the IPv6 traffic will not employ the VPN connection, and will be sent in the clear on the local network.

This could inadvertently expose sensitive traffic that was assumed to be secured by the VPN software. In this particular scenario, the resulting VPN traffic leakage is a side-effect of employing IPv6-unaware software in a dual-stacked host/network.

5. VPN traffic-leakage attacks

A local attacker could deliberately trigger IPv6 connectivity on the victim host by sending forged ICMPv6 Router Advertisement messages. Such packets could be sent by employing standard software such as rtadvd [RTADVDD], or by employing packet-crafting tools such as the [SI6-Toolkit] or THC-IPv6 [THC-IPv6]. Once IPv6 connectivity has been enabled, communications with dual-stacked systems could result in VPN traffic leakages, as previously mentioned.

While this attack may be useful enough (due to the increasing number of IPv6-enabled sites), it will only lead to traffic leakages when the destination system is dual-stacked. However, it is usually trivial for an attacker to trigger such VPN leakages for any destination systems: an attacker could simply advertise himself as the local recursive DNS server by sending forged Router Advertisement messages that include the corresponding RDNSS option, and then perform a DNS spoofing attack such that he can become a "Man in the Middle" and intercept the corresponding traffic. As with the previous attack scenario, packet-crafting tools such as [SI6-Toolkit] and [THC-IPv6] can readily perform this attack.

Some systems are known to prefer IPv6-based recursive DNS servers over IPv4-based ones, and hence the "malicious" recursive DNS servers would be preferred over the legitimate ones advertised by the VPN server.

6. Mitigations to VPN traffic-leakage vulnerabilities

There are a number of possible mitigations for the VPN traffic-leakage vulnerability discussed in this document.

If the VPN client is configured by administrative decision to redirect all traffic for IPv4 to the VPN, it should:

1. If IPv6 is not supported, disable IPv6 support in all network interfaces

For IPv6-unaware VPN clients, the most simple mitigation (although not necessarily the most desirable one) would be to disable IPv6 support in all network interface cards when a VPN connection is meant to be employed. Thus, applications on the host running the VPN client software will have no other option than to employ IPv4, and hence they will simply not even try to send/process IPv6 traffic.

2. If IPv6 is supported, ensure that all IPv6 traffic is also sent via the VPN

If the VPN client is configured to only send a subset of IPv4 networks to the VPN tunnel (split-tunnel mode), and the VPN client does not support IPv6, it should disable IPv6 as well. If it supports IPv6, it is the administrators responsibility to ensure that the correct corresponding sets of IPv4 and IPv6 networks get routed into the VPN tunnel.

Additionally, VPN clients that support IPv6 should mitigate all ND-based attacks that may introduce new entries in the routing table, such attacks based on forged RA messages containing more specific routes, forged ICMPv6 Redirect messages, etc.

A network may prevent local attackers from successfully performing the aforementioned attacks against other local hosts by implementing First-Hop Security solutions such as Router Advertisement Guard (RA-Guard) [RFC6105] and DHCPv6-Shield [I-D.gont-opsec-dhcpv6-shield]. However, for obvious reasons, a host cannot and should not rely on this type of mitigations when connecting to an open network (cybercafe, etc.).

Besides, popular implementations of RA-Guard are known to be vulnerable to evasion attacks [I-D.ietf-v6ops-ra-guard-implementation].

7. IANA Considerations

This document has no actions for IANA.

8. Security Considerations

This document discusses how traffic meant to be transferred over a VPN connection can leak out of the VPN, and hence appear in the clear on the local network. This is the result of employing IPv6-unaware VPN client software on dual-stacked hosts.

Possible ways to mitigate this problem include fixing the VPN client software, or disabling IPv6 connectivity on all network interfaces when the previous option is not feasible.

9. Acknowledgements

The author would like to thank (in alphabetical order) Gert Doering and Tor Houghton, who providing comments on earlier versions of this document.

This documents has benefited from the input of Cameron Byrne, Gert Doering, Seth Hall, Tor Houghton, Alastair Johnson, Henrik Lund Kramshoj, and Jim Small, while discussing this topic on the ipv6hackers mailing-list [IPv6-Hackers]. It has also benefited from discussions with Andrew Yourtchenko on the opsec wg mailing-list [OPSEC-LIST].

10. References

10.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC4191] Draves, R. and D. Thaler, "Default Router Preferences and More-Specific Routes", RFC 4191, November 2005.
- [RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, September 2007.
- [RFC6724] Thaler, D., Draves, R., Matsumoto, A., and T. Chown, "Default Address Selection for Internet Protocol Version 6 (IPv6)", RFC 6724, September 2012.
- [RFC6555] Wing, D. and A. Yourtchenko, "Happy Eyeballs: Success with Dual-Stack Hosts", RFC 6555, April 2012.

10.2. Informative References

- [RFC6105] Levy-Abegnoli, E., Van de Velde, G., Popoviciu, C., and J. Mohacsi, "IPv6 Router Advertisement Guard", RFC 6105, February 2011.
- [I-D.ietf-v6ops-ra-guard-implementation]
Gont, F., "Implementation Advice for IPv6 Router Advertisement Guard (RA-Guard)",
draft-ietf-v6ops-ra-guard-implementation-04 (work in progress), May 2012.
- [I-D.gont-opsec-dhcpv6-shield]
Gont, F., "DHCPv6-Shield: Protecting Against Rogue DHCPv6 Servers", draft-gont-opsec-dhcpv6-shield-00 (work in progress), May 2012.
- [IPv6-Hackers]
"IPv6 Hackers mailing-list",
<http://lists.si6networks.com/listinfo/ipv6hackers/>.
- [OPSEC-LIST]
"OPSEC WG mailing-list",
<https://www.ietf.org/mailman/listinfo/opsec>.
- [SI6-Toolkit]
"SI6 Networks' IPv6 toolkit",

<<http://www.sisnetworks.com/tools/ipv6toolkit>>.

[THC-IPv6]

"The Hacker's Choice IPv6 Attack Toolkit",
<<http://www.thc.org/thc-ipv6/>>.

[RTADVD]

"rtadvd(8) manual page", <<http://www.freebsd.org/cgi/man.cgi?query=rtadvd&sektion=8>>.

Author's Address

Fernando Gont
Huawei Technologies
Evaristo Carriego 2644
Haedo, Provincia de Buenos Aires 1706
Argentina

Phone: +54 11 4650 8472
Email: fgont@si6networks.com
URI: <http://www.si6networks.com>

OPsec Working Group
Internet-Draft
Intended status: Informational
Expires: March 29, 2015

M. Behringer
E. Vyncke
Cisco
September 25, 2014

Using Only Link-Local Addressing Inside an IPv6 Network
draft-ietf-opsec-lla-only-11

Abstract

In an IPv6 network it is possible to use only link-local addresses on infrastructure links between routers. This document discusses the advantages and disadvantages of this approach to help the decision process for a given network.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on March 29, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Using Link-Local Addressing on Infrastructure Links	2
2.1. The Approach	3
2.2. Advantages	4
2.3. Caveats	5
2.4. Internet Exchange Points	6
2.5. Summary	7
3. Security Considerations	7
4. IANA Considerations	8
5. Acknowledgements	8
6. Informative References	8
Authors' Addresses	10

1. Introduction

An infrastructure link between a set of routers typically does not require global or unique local addresses [RFC4193]. Using only link-local addressing on such links has a number of advantages. For example, that routing tables do not need to carry link addressing, and can therefore be significantly smaller. This helps to decrease failover times in certain routing convergence events. An interface of a router is also not reachable beyond the link boundaries, therefore reducing the attack surface.

This document discusses the advantages and caveats of this approach.

Note that some traditionally used techniques to operate a network such as pinging interfaces, or seeing interface information in a traceroute do not work with this approach. Details are discussed below.

During WG and IETF last call the technical correctness of the document has been reviewed, however debate exists as to whether to recommend this technique. The deployment of this technique is appropriate where it is found to be necessary.

2. Using Link-Local Addressing on Infrastructure Links

This document discusses the approach of using only link-local addresses (LLA) on all router interfaces on infrastructure links. Routers don't typically need to receive packets from hosts or nodes outside the network. For a network operator, there may be reasons to use greater than link-local scope addresses on infrastructure interfaces for certain operational tasks, such as pings to an interface or traceroutes across the network. This document discusses such cases and proposes alternative procedures.

2.1. The Approach

In this approach neither globally routed IPv6 addresses nor unique local addresses are configured on infrastructure links. In the absence of specific global or unique local address definitions, the default behavior of routers is to use link-local addresses notably for routing protocols.

The sending of ICMPv6 [RFC4443] error messages (packet-too-big, time-exceeded...) is required for routers. Therefore, another interface must be configured with an IPv6 address with a greater scope than link-local. This address will usually be a loopback interface with a global scope address belonging to the operator and part of an announced prefix (with a suitable prefix length) to avoid being dropped by other routers implementing [RFC3704]. This is implementation dependent. For the remainder of this document we will refer to this interface as a "loopback interface".

[RFC6724] recommends that greater than link-local scope IPv6 addresses are used as the source IPv6 address for all generated ICMPv6 messages sent to a non-link-local address, with the exception of ICMPv6 redirect messages, as defined in [RFC4861] section 4.5.

The effect on specific traffic types is as follows:

- o Most control plane protocols, such as BGP [RFC4271], ISIS [IS-IS], OSPFv3 [RFC5340], RIPng [RFC2080], PIM [RFC4609] work by default or can be configured to work with link-local addresses. Exceptions are explained in the caveats section (Section 2.3).
- o Management plane traffic, such as SSH [RFC4251], Telnet [RFC0495], SNMP [RFC1157], and ICMPv6 echo request [RFC4443], can use the address of the router loopback interface as the destination address. Router management can also be done over out-of-band channels.
- o ICMP error messages are usually sourced from a loopback interface with a greater than link-local address scope. [RFC4861] section 4.5 explains one exception: ICMP redirect messages can also be sourced from a link-local address.
- o Data plane traffic is forwarded independently of the link address type.
- o Neighbor discovery (neighbor solicitation and neighbor advertisement) is done by using link-local unicast and multicast addresses. Therefore neighbor discovery is not affected.

We therefore conclude that it is possible to construct a working network in this way.

2.2. Advantages

The following list of advantages is in no particular order.

Smaller routing tables: Since the routing protocol only needs to carry one global address (the loopback interface) per router, it is smaller than the traditional approach where every infrastructure link address is carried in the routing protocol. This reduces memory consumption, and increases the convergence speed in some routing failover cases. Because the Forwarding Information Base to be downloaded to line cards is smaller and there are fewer prefixes in the Routing Information Base, the routing algorithm is accelerated. Note: smaller routing tables can also be achieved by putting interfaces in passive mode for the Interior Gateway Protocol (IGP).

Simpler address management: Only loopback interface addresses need to be considered in an addressing plan. This also allows for easier renumbering.

Lower configuration complexity: link-local addresses require no specific configuration, thereby lowering the complexity and size of router configurations. This also reduces the likelihood of configuration mistakes.

Simpler DNS: Less routable address space in use also means less reverse and forward mapping DNS resource records to maintain. Of course, if the operator selects not to enter any global interface addresses in the DNS anyway, then this is less of an advantage.

Reduced attack surface: Every routable address on a router constitutes a potential attack point: a remote attacker can send traffic to that address, for example a TCP SYN flood (see [RFC4987]). If a network only uses the addresses of the router loopback interface(s), only those addresses need to be protected from outside the network. This may ease protection measures, such as infrastructure access control lists (iACL). Without using link-local addresses, it is still possible to achieve the simple iACL if the network addressing scheme is set up such that all link and loopback interfaces have greater than link-local addresses and are aggregatable, and if the infrastructure access list covers that entire aggregated space. See also [RFC6752] for further discussion on this topic. [RFC6860] describes another approach to hide addressing on infrastructure links for OSPFv2 and OSPFv3, by modifying the existing protocols. This document does not modify any protocol, however it works only for IPv6.

2.3. Caveats

The caveats listed in this section are in no particular order.

Interface ping: if an interface doesn't have a routable address, it can only be pinged from a node on the same link. Therefore, it is not possible to ping a specific link interface remotely. A possible workaround is to ping the loopback address of a router instead. In most cases today, it is not possible to see which link the packet was received on; however, [RFC5837] suggests including the interface identifier of the interface a packet was received on in the ICMPv6 response; it must be noted that there are few implementations of this ICMPv6 extension. With this approach it would be possible to ping a router on the addresses of loopback interfaces, yet see which interface the packet was received on. To check liveness of a specific interface, it may be necessary to use other methods, such as connecting to the router via SSH and checking locally or using SNMP.

Traceroute: similar to the ping case, a reply to a traceroute packet would come from the address of a loopback interface, and current implementations do not display the specific interface the packets came in on. Also here, [RFC5837] provides a solution. As in the ping case above, it is not possible to traceroute to a particular interface if it only has a link-local address. Conversely, this approach may make network topology discovery from outside the network simpler; because instead of responding with multiple different interface IP addresses, which have to be correlated by the outsider, a router will always respond with the same loopback address. If reverse DNS mapping is used, the mapping is trivial in either case.

Hardware dependency: LLAs have usually been EUI-64 based, hence, they change when the MAC address is changed. This could pose problem in a case where the routing neighbor must be configured explicitly (e.g. BGP) and a line card needs to be physically replaced hence changing the EUI-64 LLA and breaking the routing neighborship. LLAs can be statically configured such as fe80::1 and fe80::2 which can be used to configure any required static routing neighborship. However, this static LLA configuration may be more complex to operate than statically configured greater than link-local scope addresses, because LLAs are inherently ambiguous for a multi-link node such as a router; to deal with the ambiguity, the link zone index must also be considered explicitly, e.g., using the extended textual notation described in [RFC4007] as in this example: 'BGP neighbor fe80::1%eth0 is down'.

Network Management System (NMS) toolkits: if there is any NMS tool that makes use of interface IP address of a router to carry out any of its NMS functions, then it would no longer work if the interface

does not have a routable address. A possible workaround for such tools is to use the routable address of the router loopback interface instead. Most vendor implementations allow the specification of loopback interface addresses for SYSLOG, IPfix, and SNMP. The protocol LLDP (IEEE 802.1AB-2009) runs directly over Ethernet and does not require any IPv6 address, so dynamic network discovery is not hindered when using LLDP. But, network discovery based on NDP cache content will only display the link-local addresses and not the addresses of the loopback interfaces; therefore, network discovery should rather be based on the Route Information Base to detect adjacent nodes.

MPLS and RSVP-TE [RFC3209] allow establishing an MPLS LSP on a path that is explicitly identified by a strict sequence of IP prefixes or addresses (each pertaining to an interface or a router on the path). This is commonly used for Fast Re-Route (FRR). However, if an interface uses only a link-local address, then such LSPs cannot be established. At the time of writing this document, there is no workaround for this case; therefore, where RSVP-TE is being used, the approach described in this document does not work.

2.4. Internet Exchange Points

Internet Exchange Points (IXPs) have a special importance in the global Internet, because they connect a high number of networks in a single location, and because a significant part of Internet traffic passes through at least one IXP. An IXP requires therefore a very high level of security. The address space used on an IXP is generally known, as it is registered in the global Internet Route Registry, or it is easily discoverable through traceroute. The IXP prefix is especially critical, because practically all addresses on this prefix are critical systems in the Internet.

Apart from general device security guidelines, there are generally two additional ways to raise security (see also [I-D.ietf-opsec-bgp-security]):

1. Not to announce the prefix in question, and
2. To drop all traffic from remote locations destined to the IXP prefixes.

Not announcing the prefix of the IXP would frequently result in traceroute and similar packets (required for PMTUD) to be dropped due to unicast Reverse Path Forwarding (uRPF) checks. Given that PMTUD is critical, this is generally not acceptable. Dropping all external traffic to the IXP prefix is hard to implement, because if only one service provider connected to an IXP does not filter correctly, then

all IXP routers are reachable from at least that service provider network.

As the prefix used in the IXP is usually longer than a /48, it is frequently dropped by route filters on the Internet having the same net effect as not announcing the prefix.

Using link-local addresses on the IXP may help in this scenario. In this case, the generated ICMPv6 packets would be generated from loopback interfaces or from any other interface with a globally routable address without any configuration. However in this case, each service provider would use his own address space, making a generic attack against all devices on the IXP harder. All of an IXP's loopback interface addresses can be discovered by a potential attacker with a simple traceroute; a generic attack is therefore still possible, but it would require more work.

In some cases service providers carry the IXP addresses in their IGP for certain forms of traffic engineering across multiple exit points. Link-local addresses cannot be used for this purpose; in this case, the service provider would have to employ other methods of traffic engineering.

If an Internet Exchange Point is using a global prefix registered for this purpose, a traceroute will indicate whether the trace crosses an IXP rather than a private interconnect. If link local addressing is used instead, a traceroute will not provide this distinction.

2.5. Summary

Using exclusively link-local addressing on infrastructure links has a number of advantages and disadvantages, which are both described in detail in this document. A network operator can use this document to evaluate whether using link-local addressing on infrastructure links is a good idea in the context of his/her network or not. This document makes no particular recommendation either in favour or against.

3. Security Considerations

Using only LLAs on infrastructure links reduces the attack surface of a router: loopback interfaces with routed addresses are still reachable and must be secured, but infrastructure links can only be attacked from the local link. This simplifies security of control and management planes. The approach does not impact the security of the data plane. The link-local-only approach does not address control plane [RFC6192] attacks generated by data plane packets (such

as hop-limit expiration or packets containing a hop-by-hop extension header).

For additional security considerations, as previously stated, see also [RFC5837] and [I-D.ietf-opsec-bgp-security].

4. IANA Considerations

There are no IANA considerations or implications that arise from this document.

5. Acknowledgements

The authors would like to thank Salman Asadullah, Brian Carpenter, Bill Cervený, Benoit Claise, Rama Darbha, Simon Eng, Wes George, Fernando Gont, Jen Linkova, Harald Michl, Janos Mohacsi, Ivan Pepelnjak, Alvaro Retana, Jinmei Tatuya and Peter Yee for their useful comments about this work.

6. Informative References

- [I-D.ietf-opsec-bgp-security] Durand, J., Pepelnjak, I., and G. Doering, "BGP operations and security", draft-ietf-opsec-bgp-security-05 (work in progress), August 2014.
- [IS-IS] ISO/IEC 10589, , "Intermediate System to Intermediate System Intra-Domain Routing Exchange Protocol for use in Conjunction with the Protocol for Providing the Connectionless-mode Network Service (ISO 8473)", June 1992.
- [RFC0495] McKenzie, A., "Telnet Protocol specifications", RFC 495, May 1973.
- [RFC1157] Case, J., Fedor, M., Schoffstall, M., and J. Davin, "Simple Network Management Protocol (SNMP)", STD 15, RFC 1157, May 1990.
- [RFC2080] Malkin, G. and R. Minnear, "RIPng for IPv6", RFC 2080, January 1997.
- [RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, December 2001.
- [RFC3704] Baker, F. and P. Savola, "Ingress Filtering for Multihomed Networks", BCP 84, RFC 3704, March 2004.

- [RFC4007] Deering, S., Haberman, B., Jinmei, T., Nordmark, E., and B. Zill, "IPv6 Scoped Address Architecture", RFC 4007, March 2005.
- [RFC4193] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast Addresses", RFC 4193, October 2005.
- [RFC4251] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Protocol Architecture", RFC 4251, January 2006.
- [RFC4271] Rekhter, Y., Li, T., and S. Hares, "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, January 2006.
- [RFC4443] Conta, A., Deering, S., and M. Gupta, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", RFC 4443, March 2006.
- [RFC4609] Savola, P., Lehtonen, R., and D. Meyer, "Protocol Independent Multicast - Sparse Mode (PIM-SM) Multicast Routing Security Issues and Enhancements", RFC 4609, October 2006.
- [RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, September 2007.
- [RFC4987] Eddy, W., "TCP SYN Flooding Attacks and Common Mitigations", RFC 4987, August 2007.
- [RFC5340] Coltun, R., Ferguson, D., Moy, J., and A. Lindem, "OSPF for IPv6", RFC 5340, July 2008.
- [RFC5837] Atlas, A., Bonica, R., Pignataro, C., Shen, N., and JR. Rivers, "Extending ICMP for Interface and Next-Hop Identification", RFC 5837, April 2010.
- [RFC6192] Dugal, D., Pignataro, C., and R. Dunn, "Protecting the Router Control Plane", RFC 6192, March 2011.
- [RFC6724] Thaler, D., Draves, R., Matsumoto, A., and T. Chown, "Default Address Selection for Internet Protocol Version 6 (IPv6)", RFC 6724, September 2012.
- [RFC6752] Kirkham, A., "Issues with Private IP Addressing in the Internet", RFC 6752, September 2012.
- [RFC6860] Yang, Y., Retana, A., and A. Roy, "Hiding Transit-Only Networks in OSPF", RFC 6860, January 2013.

Authors' Addresses

Michael Behringer
Cisco
Building D, 45 Allee des Ormes
Mougins 06250
France

Email: mbehring@cisco.com

Eric Vyncke
Cisco
De Kleetlaan, 6A
Diegem 1831
Belgium

Email: evyncke@cisco.com

OPSEC
Internet-Draft
Intended status: Informational
Expires: November 7, 2021

E. Vyncke
Cisco
K. Chittimaneni
Square
M. Kaeo
Double Shot Security
E. Rey
ERNW
May 6, 2021

Operational Security Considerations for IPv6 Networks
draft-ietf-opsec-v6-27

Abstract

Knowledge and experience on how to operate IPv4 networks securely is available: whether it is an Internet Service Provider or an enterprise internal network. However, IPv6 presents some new security challenges. RFC 4942 describes security issues in the protocol, but network managers also need a more practical, operations-minded document to enumerate advantages and/or disadvantages of certain choices.

This document analyzes the operational security issues associated with several types of network and proposes technical and procedural mitigation techniques. This document is only applicable to managed networks, such as enterprise networks, service provider networks, or managed residential networks.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on November 7, 2021.

Copyright Notice

Copyright (c) 2021 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
1.1. Applicability Statement	4
2. Generic Security Considerations	4
2.1. Addressing	4
2.1.1. Use of ULAs	5
2.1.2. Point-to-Point Links	5
2.1.3. Loopback Addresses	5
2.1.4. Stable Addresses	6
2.1.5. Temporary Addresses for SLAAC	6
2.1.6. DHCP Considerations	8
2.1.7. DNS Considerations	8
2.1.8. Using a /64 per host	8
2.1.9. Privacy consideration of Addresses	8
2.2. Extension Headers	9
2.2.1. Order and Repetition of Extension Headers	9
2.2.2. Hop-by-Hop Options Header	10
2.2.3. Fragment Header	10
2.2.4. IP Security Extension Header	10
2.3. Link-Layer Security	11
2.3.1. Neighbor Solicitation Rate-Limiting	11
2.3.2. Router and Neighbor Advertisements Filtering	12
2.3.3. Securing DHCP	13
2.3.4. 3GPP Link-Layer Security	14
2.3.5. Impact of Multicast Traffic	15
2.3.6. SeND and CGA	15
2.4. Control Plane Security	16
2.4.1. Control Protocols	17
2.4.2. Management Protocols	18
2.4.3. Packet Exceptions	18
2.5. Routing Security	19
2.5.1. BGP Security	20

2.5.2.	Authenticating OSPFv3 Neighbors	20
2.5.3.	Securing Routing Updates	21
2.5.4.	Route Filtering	21
2.6.	Logging/Monitoring	21
2.6.1.	Data Sources	23
2.6.2.	Use of Collected Data	26
2.6.3.	Summary	29
2.7.	Transition/Coexistence Technologies	29
2.7.1.	Dual Stack	30
2.7.2.	Encapsulation Mechanisms	31
2.7.3.	Translation Mechanisms	35
2.8.	General Device Hardening	37
3.	Enterprises Specific Security Considerations	37
3.1.	External Security Considerations	38
3.2.	Internal Security Considerations	39
4.	Service Providers Security Considerations	40
4.1.	BGP	40
4.1.1.	Remote Triggered Black Hole Filtering (RTBH)	40
4.2.	Transition/Coexistence Mechanism	40
4.3.	Lawful Intercept	40
5.	Residential Users Security Considerations	41
6.	Further Reading	41
7.	Acknowledgements	42
8.	Security Considerations	42
9.	References	42
9.1.	Normative References	42
9.2.	Informative References	42
	Authors' Addresses	57

1. Introduction

Running an IPv6 network is new for most operators not only because they are not yet used to large-scale IPv6 networks but also because there are subtle but critical and important differences between IPv4 and IPv6, especially with respect to security. For example, all layer-2 interactions are now done using Neighbor Discovery Protocol [RFC4861] rather than using Address Resolution Protocol [RFC0826]. Also, there is no Network Address Port Translation (NAPT) defined in [RFC2663] for IPv6 even if [RFC6296] specifies a Network Prefix Translation for IPv6 (NPTv6) which is a 1-to-1 mapping of IPv6 addresses. Another important difference is that IPv6 is extensible with the use of extension headers.

IPv6 networks are deployed using a variety of techniques, each of which have their own specific security concerns.

This document complements [RFC4942] by listing security issues when operating a network (including various transition technologies). It

also provides more recent operational deployment experiences where warranted.

1.1. Applicability Statement

This document is applicable to managed networks, i.e., when the network is operated by the user organization itself. Indeed, many of the recommended mitigation techniques must be configured with detailed knowledge of the network (which are the default routers, the switch trunk ports, etc.). This covers Service Provider (SP), enterprise networks and some knowledgeable-home-user-managed residential networks. This applicability statement especially applies to Section 2.3 and Section 2.5.4.

2. Generic Security Considerations

2.1. Addressing

IPv6 address allocations and overall architecture are an important part of securing IPv6. Initial designs, even if intended to be temporary, tend to last much longer than expected. Although initially IPv6 was thought to make renumbering easy, in practice it may be extremely difficult to renumber without a proper IP Address Management (IPAM) system. [RFC7010] introduces the mechanisms that could be utilized for IPv6 site renumbering and tries to cover most of the explicit issues and requirements associated with IPv6 renumbering.

A key task for a successful IPv6 deployment is to prepare an addressing plan. Because an abundance of address space is available, structuring an address plan around both services and geographic locations allows address space to become a basis for more structured security policies to permit or deny services between geographic regions. [RFC6177] documents some operational considerations of using different prefix sizes for address assignments at end sites.

A common question is whether companies should use Provider Independent (PI) vs. Provider Allocated (PA) space [RFC7381], but from a security perspective there is little difference. However, one aspect to keep in mind is who has administrative ownership of the address space and who is technically responsible if/when there is a need to enforce restrictions on routability of the space, e.g., due to malicious criminal activity originating from it. Relying on PA address space may also increase the perceived need for address translation techniques such as NPTv6 and thereby augmenting the complexity of the operations including the security operations.

In [RFC7934], it is recommended that IPv6 network deployments provide multiple IPv6 addresses from each prefix to general-purpose hosts and it specifically does not recommend limiting a host to only one IPv6 address per prefix. It also recommends that the network give the host the ability to use new addresses without requiring explicit requests (for example by using SLAAC). Privacy Extensions as of [RFC8981] constitute one of the main scenarios where hosts are expected to generate multiple addresses from the same prefix and having multiple IPv6 addresses per interface is a major change compared to the unique IPv4 address per interface for hosts (secondary IPv4 addresses are not common); especially for audits (see section Section 2.6.2.3).

2.1.1. Use of ULAs

Unique Local Addresses (ULAs) [RFC4193] are intended for scenarios where interfaces are not globally reachable, despite being routed within a domain. They formally have global scope, but [RFC4193] specifies that they must be filtered at domain boundaries. ULAs are different from [RFC1918] addresses and have different use cases. One use of ULA is described in [RFC4864], another one is for internal communication stability in networks where external connectivity may come and go (e.g., some ISPs provide ULAs in home networks connected via a cable modem). It should further be kept in mind that ULA /48s from the fd00::/8 space (L=1) MUST be generated with a pseudo-random algorithm, per [RFC4193] section 3.2.1.

2.1.2. Point-to-Point Links

[RFC6164] in section 5.1 specifies the rationale of using /127 for inter-router point-to-point links to prevent the ping-pong issue between routers not correctly implementing [RFC4443] and also prevents a DoS attack on the neighbor cache. The previous recommendation of [RFC3627] has been obsoleted and marked Historic by [RFC6547]).

Some environments are also using link-local addressing for point-to-point links. While this practice could further reduce the attack surface of infrastructure devices, the operational disadvantages also need to be carefully considered; see also [RFC7404].

2.1.3. Loopback Addresses

Many operators reserve a /64 block for all loopback addresses in their infrastructure and allocate a /128 out of this reserved /64 prefix for each loopback interface. This practice facilitates configuration of Access Control List (ACL) rules to enforce a security policy for those loopback addresses.

2.1.4. Stable Addresses

When considering how to assign stable addresses for nodes (either by static configuration or by pre-provisioned DHCPv6 lease Section 2.1.6), it is necessary to take into consideration the effectiveness of perimeter security in a given environment.

There is a trade-off between ease of operation (where some portions of the IPv6 address could be easily recognizable for operational debugging and troubleshooting) versus the risk of trivial scanning used for reconnaissance. [SCANNING] shows that there are scientifically based mechanisms that make scanning for IPv6 reachable nodes more feasible than expected; see also [RFC7707].

Stable addresses also allow easy enforcement of a security policy at the perimeter based on IPv6 addresses. E.g., Manufacturer Usage Description (MUD) [RFC8520] is a mechanism where the perimeter defense can retrieve security policy template based on the type of internal device and apply the right security policy based on the device IPv6 address.

The use of well-known IPv6 addresses (such as ff02::1 for all link-local nodes) or the use of commonly repeated addresses could make it easy to figure out which devices are name servers, routers, or other critical devices; even a simple traceroute will expose most of the routers on a path. There are many scanning techniques possible and operators should not rely on the 'impossible to find because my address is random' paradigm (a.k.a. "security by obscurity"), even if it is common practice to have the stable addresses randomly distributed across /64 subnets and to always use DNS (as IPv6 addresses are hard for human brains to remember).

While in some environments obfuscating addresses could be considered an added benefit, it should not preclude enforcement of perimeter rules. Stable addresses following some logical allocation scheme may ease the operation (as simplicity always helps security).

Typical deployments will have a mix of stable and non-stable addresses; the stable addresses being either predictable (e.g., ::25 for a mail server) or obfuscated (i.e., appearing as a random 64-bit number).

2.1.5. Temporary Addresses for SLAAC

Historically, stateless address autoconfiguration (SLAAC) makes up the globally unique IPv6 address based on an automatically generated 64-bit interface identifier (IID) based on the EUI-64 MAC address combined with the /64 prefix (received in the Prefix Information

Option (PIO) of the Router Advertisement (RA)). The EUI-64 address is generated from the stable 48-bit MAC address and does not change even if the host moves to another network; this is of course bad for privacy as a host can be traced from network (home) to network (office or Wi-Fi in hotels). [RFC8064] recommends against the use of EUI-64 addresses; and it must be noted that most host operating systems do not use EUI-64 addresses anymore and rely on either [RFC8981] or [RFC8064].

Randomly generating an interface ID, as described in [RFC8981], is part of SLAAC with so-called privacy extension addresses and is used to address some privacy concerns. Privacy extension addresses, a.k.a., temporary addresses may help to mitigate the correlation of activities of a node within the same network and may also reduce the attack exposure window. But using [RFC8981] privacy extension addresses might prevent the operator from building host specific access control lists (ACLs). The [RFC8981] privacy extension addresses could also be used to obfuscate some malevolent activities and specific user attribution/accountability procedures should be put in place as described in Section 2.6.

[RFC8064] combined with the address generation mechanism of [RFC7217] specifies another way to generate an address while still keeping the same IID for each network prefix; this allows SLAAC nodes to always have the same stable IPv6 address on a specific network while having different IPv6 addresses on different networks.

In some specific use cases where user accountability is more important than user privacy, network operators may consider disabling SLAAC and relying only on DHCPv6; but not all operating systems support DHCPv6 so some hosts will not get any IPv6 connectivity. Disabling SLAAC and privacy extension addresses can be done for most operating systems by sending RA messages with a hint to get addresses via DHCPv6 by setting the M-bit and disabling SLAAC by resetting all A-bits in all prefix information options. However, attackers could still find ways to bypass this mechanism if not enforced at the switch/router level.

However, in scenarios where anonymity is a strong desire (protecting user privacy is more important than user attribution), privacy extension addresses should be used. When mechanisms recommended by [RFC8064] are available, the stable privacy address is probably a good balance between privacy (among different networks) and security/user attribution (within a network).

2.1.6. DHCP Considerations

Some environments use DHCPv6 to provision addresses and other parameters in order to ensure auditability and traceability (see Section 2.6.1.5 for the limitations of DHCPv6 for auditability).

A main security concern is the ability to detect and counteract rogue DHCP servers (Section 2.3.3). It must be noted that as opposed to DHCPv4, DHCPv6 can lease several IPv6 addresses per client. For DHCPv4, the lease is bound to the 'client identifier', which may contain a hardware address, or it may contain another type of identifier, such as a DNS name. For DHCPv6, the lease is bound to the client DHCP Unique ID (DUID), which may, or may not, be bound to the client link-layer address. [RFC7824] describes the privacy issues associated with the use of DHCPv6 by Internet users. The anonymity profiles [RFC7844] are designed for clients that wish to remain anonymous to the visited network. [RFC7707] recommends that DHCPv6 servers issue addresses randomly from a large pool.

2.1.7. DNS Considerations

While the security concerns of DNS are not fundamentally different between IPv4 and IPv6, there are specific considerations in DNS64 [RFC6147] environments that need to be understood. Specifically, the interactions and the potential of interference with DNSSEC ([RFC4033]) implementation need to be understood - these are pointed out in more detail in Section 2.7.3.2.

2.1.8. Using a /64 per host

An interesting approach is using a /64 per host as proposed in [RFC8273] especially in a shared environment. This allows for easier user attribution (typically based on the host MAC address) as its /64 prefix is stable even if applications within the host can change their IPv6 address within this /64 prefix.

This can also be useful for the generation of ACLs once individual systems (e.g. admin workstations) have their own prefixes.

2.1.9. Privacy consideration of Addresses

Beside the security aspects of IPv6 addresses, there are also privacy considerations: mainly because they are of global scope and visible globally. [RFC7721] goes into more detail on the privacy considerations for IPv6 addresses by comparing the manually configured IPv6 address, DHCPv6, and SLAAC.

2.2. Extension Headers

Extension headers are an important difference between IPv4 and IPv6. In IPv4-based packets, it's trivial to find the upper-layer protocol type and protocol header, while in IPv6 it is more complex since the extension header chain must be parsed completely (even if not processed) in order to find the upper-layer protocol header. IANA has closed the existing empty "Next Header Types" registry to new entries and is redirecting its users to a new "IPv6 Extension Header Types" registry per [RFC7045].

Extension headers have also become a very controversial topic since forwarding nodes that discard packets containing extension headers are known to cause connectivity failures and deployment problems [RFC7872]. Understanding the role of various extension headers is important and this section enumerates the ones that need careful consideration.

A clarification on how intermediate nodes should handle packets with existing or future extension headers is found in [RFC7045]. The uniform TLV format to be used for defining future extension headers is described in [RFC6564]. Sections 5.2 and 5.3 of [RFC8504] provide more information on the processing of extension headers by IPv6 nodes.

Vendors of filtering solutions and operations personnel responsible for implementing packet filtering rules should be aware that the 'Next Header' field in an IPv6 header can both point to an IPv6 extension header or to an upper layer protocol header. This has to be considered when designing the user interface of filtering solutions or during the creation of filtering rule sets.

There is IETF work in progress regarding filtering rules for those extension headers: [I-D.ietf-opsec-ipv6-eh-filtering] for transit routers.

2.2.1. Order and Repetition of Extension Headers

While [RFC8200] recommends the order and the maximum repetition of extension headers, there are still IPv6 implementations, at the time of writing, which support a non-recommended order of headers (such as ESP before routing) or an illegal repetition of headers (such as multiple routing headers). The same applies for options contained in the extension headers (see [I-D.kampanakis-6man-ipv6-eh-parsing]). In some cases, it has led to nodes crashing when receiving or forwarding wrongly formatted packets.

A firewall or edge device should be used to enforce the recommended order and the maximum occurrences of extension headers by dropping non-conforming packets.

2.2.2. Hop-by-Hop Options Header

In the previous IPv6 specification [RFC2460], the hop-by-hop options header, when present in an IPv6 packet, forced all nodes to inspect and possibly process this header. This enabled denial-of-service attacks as most, if not all, routers cannot process this type of packet in hardware but have to process these packets in software and hence compete with other software tasks, such as handling the control and management plane processing.

Section 4.3 of the current Internet Standard for IPv6, [RFC8200], has taken this attack vector into account and made the processing of hop-by-hop options headers by intermediate routers explicitly configurable.

2.2.3. Fragment Header

The fragment header is used by the source (and only the source) when it has to fragment packets. [RFC7112] and section 4.5 of [RFC8200] explain why it is important that:

Firewall and security devices should drop first fragments that do not contain the entire IPv6 header chain (including the transport-layer header).

Destination nodes should discard first fragments that do not contain the entire IPv6 header chain (including the transport-layer header).

If those requirements are not met, stateless filtering could be bypassed by a hostile party. [RFC6980] applies a stricter rule to Neighbor Discovery Protocol (NDP) by enforcing the drop of fragmented NDP packets (except for "Certification Path Advertisement" messages as noted in section Section 2.3.2.1). [RFC7113] describes how the RA-guard function described in [RFC6105] should behave in the presence of fragmented RA packets.

2.2.4. IP Security Extension Header

The IPsec [RFC4301] extension headers (AH [RFC4302] and ESP [RFC4303]) are required if IPsec is to be utilized for network level security. Previously, IPv6 mandated implementation of IPsec but [RFC6434] updated that recommendation by making support of the IPsec

architecture [RFC4301] a SHOULD for all IPv6 nodes which is also retained in the latest IPv6 Nodes Requirement standard [RFC8504].

2.3. Link-Layer Security

IPv6 relies heavily on NDP [RFC4861] to perform a variety of link operations such as discovering other nodes on the link, resolving their link-layer addresses, and finding routers on the link. If not secured, NDP is vulnerable to various attacks, such as router/neighbor message spoofing, redirect attacks, Duplicate Address Detection (DAD) DoS attacks, etc. Many of these security threats to NDP have been documented in IPv6 ND Trust Models and Threats [RFC3756] and in [RFC6583].

Most of the issues are only applicable when the attacker is on the same link but NDP also has security issues when the attacker is off-link, see the section below Section 2.3.1.

2.3.1. Neighbor Solicitation Rate-Limiting

NDP can be vulnerable to remote denial of service (DoS) attacks; for example, when a router is forced to perform address resolution for a large number of unassigned addresses, i.e., when a prefix is scanned by an attacker in a fast manner. This can keep new devices from joining the network or render the last-hop router ineffective due to high CPU usage. Easy mitigative steps include rate-limiting Neighbor Solicitations, restricting the amount of state reserved for unresolved solicitations, and clever cache/timer management.

[RFC6583] discusses the potential for off-link DoS in detail and suggests implementation improvements and operational mitigation techniques that may be used to mitigate or alleviate the impact of such attacks. Here are some feasible mitigation options that can be employed by network operators today:

- o Ingress filtering of unused addresses by ACL. These require stable configuration of the addresses; for example, allocating the addresses out of a /120 and using a specific ACL to only allow traffic to this /120 (of course, the actual hosts are configured with a /64 prefix for the link).
- o Tuning of NDP process (where supported), e.g., enforcing limits on data structures such as the number of neighbor cache entries in 'incomplete' state (e.g., 256 incomplete entries per interface) or the rate of NA per interface (e.g., 100 NA per second).
- o Using a /127 on a point-to-point link, per [RFC6164].

- o Using only link-local addresses on links where there are only routers, see [RFC7404]

2.3.2. Router and Neighbor Advertisements Filtering

2.3.2.1. Router Advertisement Filtering

Router Advertisement spoofing is a well-known on-link attack vector and has been extensively documented. The presence of rogue RAs, either unintentional or malicious, can cause partial or complete failure of operation of hosts on an IPv6 link. For example, a node can select an incorrect router address which can then be used for an on-path attack or the node can assume wrong prefixes to be used for SLAAC. [RFC6104] summarizes the scenarios in which rogue RAs may be observed and presents a list of possible solutions to the problem. [RFC6105] (RA-Guard) describes a solution framework for the rogue RA problem where network segments are designed around switching devices that are capable of identifying invalid RAs and blocking them before the attack packets actually reach the target nodes.

However, several evasion techniques that circumvent the protection provided by RA-Guard have surfaced. A key challenge to this mitigation technique is introduced by IPv6 fragmentation. Attackers can conceal their attack by fragmenting their packets into multiple fragments such that the switching device that is responsible for blocking invalid RAs cannot find all the necessary information to perform packet filtering of the same packet. [RFC7113] describes such evasion techniques and provides advice to RA-Guard implementers such that the aforementioned evasion vectors can be eliminated.

Given that the IPv6 Fragmentation Header can be leveraged to circumvent some implementations of RA-Guard, [RFC6980] updates [RFC4861] such that use of the IPv6 Fragmentation Header is forbidden in all Neighbor Discovery messages except "Certification Path Advertisement", thus allowing for simple and effective measures to counter fragmented NDP attacks.

2.3.2.2. Neighbor Advertisement Filtering

The Source Address Validation Improvements (SAVI) working group has worked on other ways to mitigate the effects of such attacks. [RFC7513] helps in creating bindings between a DHCPv4 [RFC2131] /DHCPv6 [RFC8415] assigned source IP address and a binding anchor [RFC7039] on a SAVI device. Also, [RFC6620] describes how to glean similar bindings when DHCP is not used. The bindings can be used to filter packets generated on the local link with forged source IP addresses.

2.3.2.3. Host Isolation

Isolating hosts for the NDP traffic can be done by using a /64 per host, refer to Section 2.1.8, as NDP is only relevant within a /64 on-link prefix; 3GPP Section 2.3.4 uses a similar mechanism.

A more drastic technique to prevent all NDP attacks is based on isolation of all hosts with specific configurations. In such a scenario, hosts (i.e., all nodes that are not routers) are unable to send data-link layer frames to other hosts, therefore, no host-to-host attacks can happen. This specific setup can be established on some switches or Wi-Fi access points. This is not always feasible when hosts need to communicate with other hosts in the same subnet, e.g., for access to file shares.

2.3.2.4. NDP Recommendations

It is still recommended that RA-Guard and SAVI be employed as a first line of defense against common attack vectors including misconfigured hosts. This recommendation also applies when DHCPv6 is used, as RA messages are used to discover the default router(s) and for on-link prefix determination. This line of defense is most effective when incomplete fragments are dropped by routers and switches as described in Section 2.2.3. The generated log should also be analyzed to identify and act on violations.

Network operators should be aware that RA-Guard and SAVI do not work as expected or could even be harmful in specific network configurations (notably when there could be multiple routers).

Enabling RA-Guard by default in managed networks (e.g., Wi-Fi networks, enterprise campus networks, etc.) should be strongly considered except for specific use cases such as the presence of homenet devices emitting router advertisements.

2.3.3. Securing DHCP

The Dynamic Host Configuration Protocol for IPv6 (DHCPv6), as described in [RFC8415], enables DHCP servers to pass configuration parameters, such as IPv6 network addresses and other configuration information, to IPv6 nodes. DHCP plays an important role in most large networks by providing robust stateful configuration in the context of automated system provisioning.

The two most common threats to DHCP clients come from malicious (a.k.a., rogue) or unintentionally misconfigured DHCP servers. In these scenarios, a malicious DHCP server is established with the intent of providing incorrect configuration information to the

clients to cause a denial-of-service attack or to mount on-path attack. While unintentional, a misconfigured DHCP server can have the same impact. Additional threats against DHCP are discussed in the security considerations section of [RFC8415].

DHCPv6-Shield, [RFC7610], specifies a mechanism for protecting connected DHCPv6 clients against rogue DHCPv6 servers. This mechanism is based on DHCPv6 packet-filtering at the layer-2 device, i.e., the administrator specifies the interfaces connected to DHCPv6 servers. However, extension headers could be leveraged to bypass DHCPv6-Shield unless [RFC7112] is enforced.

It is recommended to use DHCPv6-Shield and to analyze the corresponding log messages.

2.3.4. 3GPP Link-Layer Security

The 3GPP link is a point-to-point like link that has no link-layer address. This implies there can only be one end host (the mobile hand-set) and the first-hop router (i.e., a GPRS Gateway Support Node (GGSN) or a Packet Gateway (PGW)) on that link. The GGSN/PGW never configures a non link-local address on the link using the advertised /64 prefix on it; see Section 2.1.8. The advertised prefix must not be used for on-link determination. There is no need for address resolution on the 3GPP link, since there are no link-layer addresses. Furthermore, the GGSN/PGW assigns a prefix that is unique within each 3GPP link that uses IPv6 stateless address autoconfiguration. This avoids the necessity to perform DAD at the network level for every address generated by the mobile host. The GGSN/PGW always provides an IID to the cellular host for the purpose of configuring the link-local address and ensures the uniqueness of the IID on the link (i.e., no collisions between its own link-local address and the mobile host's address).

The 3GPP link model itself mitigates most of the known NDP-related Denial-of-Service attacks. In practice, the GGSN/PGW only needs to route all traffic to the mobile host that falls under the prefix assigned to it. As there is also a single host on the 3GPP link, there is no need to defend that IPv6 address.

See Section 5 of [RFC6459] for a more detailed discussion on the 3GPP link model, NDP, and the address configuration details. In some mobile networks, DHCPv6 and DHCP-PD are also used.

2.3.5. Impact of Multicast Traffic

IPv6 uses multicast extensively for signaling messages on the local link to avoid broadcast messages for on-the-wire efficiency.

The use of multicast has some side effects on wireless networks, such as a negative impact on battery life of smartphones and other battery-operated devices that are connected to such networks. [RFC7772] and [RFC6775] (for specific wireless networks) discuss methods to rate-limit RAs and other ND messages on wireless networks in order to address this issue.

The use of link-layer multicast addresses (e.g., ff02::1 for the all nodes link-local multicast address) could also be misused for an amplification attack. Imagine, a hostile node sending an ICMPv6 ECHO_REQUEST to ff02::1 with a spoofed source address, then, all link-local nodes will reply with ICMPv6 ECHO_REPLY packets to the source address. This could be a DoS attack for the address owner. This attack is purely local to the layer-2 network as packets with a link-local destination are never forwarded by an IPv6 router.

This is the reason why large Wi-Fi network deployments often limit the use of link-layer multicast either from or to the uplink of the Wi-Fi access point, i.e., Wi-Fi stations are prevented to send link-local multicast to their direct neighboring Wi-Fi stations; this policy also blocks service discovery via mDNS ([RFC6762]) and LLmNR ([RFC4795]).

2.3.6. SeND and CGA

SEcure Neighbor Discovery (SeND), as described in [RFC3971], is a mechanism that was designed to secure ND messages. This approach involves the use of new NDP options to carry public key-based signatures. Cryptographically Generated Addresses (CGA), as described in [RFC3972], are used to ensure that the sender of a Neighbor Discovery message is the actual "owner" of the claimed IPv6 address. A new NDP option, the CGA option, was introduced and is used to carry the public key and associated parameters. Another NDP option, the RSA Signature option, is used to protect all messages relating to neighbor and Router discovery.

SeND protects against:

- o Neighbor Solicitation/Advertisement Spoofing
- o Neighbor Unreachability Detection Failure
- o Duplicate Address Detection DoS Attack

- o Router Solicitation and Advertisement Attacks
- o Replay Attacks
- o Neighbor Discovery DoS Attacks

SeND does NOT:

- o Protect statically configured addresses
- o Protect addresses configured using fixed identifiers (i.e., EUI-64)
- o Provide confidentiality for NDP communications
- o Compensate for an unsecured link - SeND does not require that the addresses on the link and Neighbor Advertisements correspond.

However, at this time and over a decade since their original specifications, CGA and SeND do not have support from widely deployed IPv6 devices; hence, their usefulness is limited and should not be relied upon.

2.4. Control Plane Security

[RFC6192] defines the router control plane and provides detailed guidance to secure it for IPv4 and IPv6 networks. This definition is repeated here for the reader's convenience. Please note that the definition is completely protocol-version agnostic (most of this section applies to IPv6 in the same way as to IPv4).

Preamble: IPv6 control plane security is vastly congruent with its IPv4 equivalent with the exception of OSPFv3 authentication (Section 2.4.1) and some packet exceptions (see Section 2.4.3) that are specific to IPv6.

Modern router architecture design maintains a strict separation of forwarding and router control plane hardware and software. The router control plane supports routing and management functions. It is generally described as the router architecture hardware and software components for handling packets destined to the device itself, as well as, building and sending packets originated locally on the device. The forwarding plane is typically described as the router architecture hardware and software components responsible for receiving a packet on an incoming interface, performing a lookup to identify the packet's IP next hop and best outgoing interface towards the destination, and forwarding the packet through the appropriate outgoing interface.

While the forwarding plane is usually implemented in high-speed hardware, the control plane is implemented by a generic processor (referred to as the route processor (RP)) and cannot process packets at a high rate. Hence, this processor can be attacked by flooding its input queue with more packets than it can process. The control plane processor is then unable to process valid control packets and the router can lose IGP or BGP adjacencies which can cause a severe network disruption.

[RFC6192] provides detailed guidance to protect the router control plane in IPv6 networks. The rest of this section contains simplified guidance.

The mitigation techniques are:

- o To drop non-legit or potentially harmful control packets before they are queued to the RP (this can be done by a forwarding plane ACL) and
- o To rate-limit the remaining packets to a rate that the RP can sustain. Protocol-specific protection should also be done (for example, a spoofed OSPFv3 packet could trigger the execution of the Dijkstra algorithm, therefore, the frequency of Dijkstra calculations should be also rate-limited).

This section will consider several classes of control packets:

- o Control protocols: routing protocols: such as OSPFv3, BGP, RIPng, and by extension NDP and ICMP
- o Management protocols: SSH, SNMP, NETCONF, RESTCONF, IPFIX, etc.
- o Packet exceptions: normal data packets that require a specific processing such as generating a packet-too-big ICMP message or processing the hop-by-hop options header.

2.4.1. Control Protocols

This class includes OSPFv3, BGP, NDP, ICMP.

An ingress ACL to be applied on all the router interfaces for packets to be processed by the RP should be configured to:

- o drop OSPFv3 (identified by Next-Header being 89) and RIPng (identified by UDP port 521) packets from a non link-local address (except for OSPFv3 virtual links)

- o allow BGP (identified by TCP port 179) packets from all BGP neighbors and drop the others
- o allow all ICMP packets (transit and to the router interfaces)

Note: dropping OSPFv3 packets which are authenticated by IPsec could be impossible on some routers that are unable to parse the IPsec ESP or AH extension headers during ACL classification.

Rate-limiting of the valid packets should be done, see also [RFC8541] for a side benefit for OSPv3. The exact configuration will depend on the available resources of the router (CPU, TCAM, ...).

2.4.2. Management Protocols

This class includes: SSH, SNMP, RESTCONF, NETCONF, gRPC, syslog, NTP, etc.

An ingress ACL to be applied on all the router interfaces (or at ingress interfaces of the security perimeter or by using specific features of the platform) should be configured for packets destined to the RP such as:

- o Drop packets destined to the routers except those belonging to protocols which are used (for example, permit TCP 22 and drop all others when only SSH is used);
- o Drop packets where the source does not match the security policy, for example, if SSH connections should only be originated from the Network Operation Center (NOC), then the ACL should permit TCP port 22 packets only from the NOC prefix.

Rate-limiting of valid packets should be done. The exact configuration will depend on the available router resources.

2.4.3. Packet Exceptions

This class covers multiple cases where a data plane packet is punted to the route processor because it requires specific processing:

- o generation of an ICMP packet-too-big message when a data plane packet cannot be forwarded because it is too large (required to discover the Path MTU);
- o generation of an ICMP hop-limit-expired message when a data plane packet cannot be forwarded because its hop-limit field has reached 0 (also used by the traceroute utility);

- o generation of an ICMP destination-unreachable message when a data plane packet cannot be forwarded for any reason;
- o processing of the hop-by-hop options header, new implementations follow section 4.3 of [RFC8200] where this processing is optional;
- o or more specific to some router implementation: an oversized extension header chain which cannot be processed by the hardware and force the packet to be punted to the RP.

On some routers, not everything can be done by the specialized data plane hardware which requires some packets to be 'punted' to the generic RP. This could include for example the processing of a long extension header chain in order to apply an ACL based on layer-4 information. [RFC6980] and more generally [RFC7112] highlight the security implications of oversized extension header chains on routers and updates the original IPv6 specifications, [RFC2460], such that the first fragment of a packet is required to contain the entire IPv6 header chain. Those changes are incorporated in the IPv6 standard [RFC8200]

An ingress ACL cannot mitigate a control plane attack using these packet exceptions. The only protection for the RP is to rate-limit those packet exceptions that are forwarded to the RP, this means that some data plane packets will be dropped without an ICMP message sent to the source which may delay Path MTU discovery and cause drops.

In addition to limiting the rate of data plane packets queued to the RP, it is also important to rate-limit the generation of ICMP messages. This is important both to preserve RP resources and also to prevent an amplification attack using the router as a reflector. It is worth noting that some platforms implement this rate-limiting in hardware. Of course, a consequence of not generating an ICMP message will break some IPv6 mechanisms such as Path MTU discovery or a simple traceroute.

2.5. Routing Security

Preamble: IPv6 routing security is congruent with IPv4 routing security with the exception of OSPv3 neighbor authentication (see Section 2.5.2).

Routing security in general can be broadly divided into three sections:

1. Authenticating neighbors/peers
2. Securing routing updates between peers

3. Route filtering

[RFC5082] is also applicable to IPv6 and can ensure that routing protocol packets are coming from the local network; it must also be noted that in IPv6 all interior gateway protocols use link-local addresses.

As for IPv4, it is recommended to enable a routing protocol only on interfaces where it is required.

2.5.1. BGP Security

As BGP is identical for IPv4 and IPv6 and as [RFC7454] covers all the security aspects for BGP in detail, [RFC7454] is also applicable to IPv6.

2.5.2. Authenticating OSPFv3 Neighbors

OSPFv3 can rely on IPsec to fulfill the authentication function. Operators should note that IPsec support is not standard on all routing platforms. In some cases, this requires specialized hardware that offloads crypto over to dedicated ASICs or enhanced software images (both of which often come with added financial cost) to provide such functionality. An added detail is to determine whether OSPFv3 IPsec implementations use AH or ESP-Null for integrity protection. In early implementations, all OSPFv3 IPsec configurations relied on AH since the details weren't specified in [RFC5340]. However, the document which specifically describes how IPsec should be implemented for OSPFv3 [RFC4552] specifically states that "ESP-Null MUST and AH MAY be implemented" since it follows the overall IPsec standards wording. OSPFv3 can also use normal ESP to encrypt the OSPFv3 payload to provide confidentiality for the routing information.

[RFC7166] changes OSPFv3 reliance on IPsec by appending an authentication trailer to the end of the OSPFv3 packets; it does not specifically authenticate the specific originator of an OSPFv3 packet; rather, it allows a router to confirm that the packet has been issued by a router that had access to the shared authentication key.

With all authentication mechanisms, operators should confirm that implementations can support re-keying mechanisms that do not cause outages. There have been instances where any re-keying causes outages and therefore, the tradeoff between utilizing this functionality needs to be weighed against the protection it provides. [RFC4107] documents some guidelines for crypto keys management.

2.5.3. Securing Routing Updates

IPv6 initially mandated the provisioning of IPsec capability in all nodes. However, in the updated IPv6 Nodes Requirement standard [RFC8504], IPsec is a 'SHOULD' and not a 'MUST' implement. Theoretically, it is possible that all communication between two IPv6 nodes, especially routers exchanging routing information, is encrypted using IPsec. In practice however, deploying IPsec is not always feasible given hardware and software limitations of the various platforms deployed.

Many routing protocols support the use of cryptography to protect the routing updates, the use of this protection is recommended; [RFC8177] is a YANG data model for key chains that includes re-keying functionality.

2.5.4. Route Filtering

Route filtering policies will be different depending on whether they pertain to edge route filtering vs. internal route filtering. At a minimum, IPv6 routing policy as it pertains to routing between different administrative domains should aim to maintain parity with IPv4 from a policy perspective, e.g.,

- o Filter internal-use, non-globally routable IPv6 addresses at the perimeter;
- o Discard routes for bogon [CYMRU] and reserved space (see [RFC8190]);
- o Configure ingress route filters that validate route origin, prefix ownership, etc. through the use of various routing databases, e.g., [RADB]. [RFC8210] formally validates the origin ASs of BGP announcements.

Some good guidance can be found at [RFC7454].

A valid routing table can also be used to apply network ingress filtering (see [RFC2827]).

2.6. Logging/Monitoring

In order to perform forensic research in the cases of a security incident or detecting abnormal behavior, network operators should log multiple pieces of information. In some cases, this requires a frequent poll of devices via a Network Management Station.

This logging should include, but not limited to:

- o logs of all applications using the network (including user space and kernel space) when available (for example web servers that the network operator manages);
- o data from IP Flow Information Export [RFC7011] also known as IPFIX;
- o data from various SNMP MIBs [RFC4293] or YANG data via RESTCONF [RFC8040] or NETCONF [RFC6241];
- o historical data of Neighbor Cache entries;
- o stateful DHCPv6 [RFC8415] lease cache, especially when a relay agent [RFC6221] is used;
- o Source Address Validation Improvement (SAVI) [RFC7039] events, especially the binding of an IPv6 address to a MAC address and a specific switch or router interface;
- o firewall ACL log;
- o authentication server log;
- o RADIUS [RFC2866] accounting records.

Please note that there are privacy issues or regulations related to how these logs are collected, stored, used, and safely discarded. Operators are urged to check their country legislation (e.g., General Data Protection Regulation GDPR [GDPR] in the European Union).

All those pieces of information can be used for:

- o forensic (Section 2.6.2.1) investigations such as who did what and when?
- o correlation (Section 2.6.2.3): which IP addresses were used by a specific node (assuming the use of privacy extensions addresses [RFC8981])
- o inventory (Section 2.6.2.2): which IPv6 nodes are on my network?
- o abnormal behavior detection (Section 2.6.2.4): unusual traffic patterns are often the symptoms of an abnormal behavior which is in turn a potential attack (denial-of-service, network scan, a node being part of a botnet, etc.)

2.6.1. Data Sources

This section lists the most important sources of data that are useful for operational security.

2.6.1.1. Application Logs

Those logs are usually text files where the remote IPv6 address is stored in clear text (not binary). This can complicate the processing since one IPv6 address, for example 2001:db8::1 can be written in multiple ways, such as:

- o 2001:DB8::1 (in uppercase)
- o 2001:0db8::0001 (with leading 0)
- o and many other ways including the reverse DNS mapping into a FQDN (which should not be trusted).

[RFC5952] explains this problem in detail and recommends the use of a single canonical format. This document recommends the use of canonical format [RFC5952] for IPv6 addresses in all possible cases. If the existing application cannot log using the canonical format, then it is recommended to use an external post-processing program in order to canonicalize all IPv6 addresses.

2.6.1.2. IP Flow Information Export by IPv6 Routers

IPFIX [RFC7012] defines some data elements that are useful for security:

- o nextHeaderIPv6, sourceIPv6Address, and destinationIPv6Address;
- o sourceMacAddress and destinationMacAddress.

The IP version is the ipVersion element defined in [IANA-IPFIX].

Moreover, IPFIX is very efficient in terms of data handling and transport. It can also aggregate flows by a key such as sourceMacAddress in order to have aggregated data associated with a specific sourceMacAddress. This memo recommends the use of IPFIX and aggregation on nextHeaderIPv6, sourceIPv6Address, and sourceMacAddress.

2.6.1.3. SNMP MIB and NETCONF/RESTCONF YANG Modules data by IPv6 Routers

RFC 4293 [RFC4293] defines a Management Information Base (MIB) for the two address families of IP. This memo recommends the use of:

- o ipIfStatsTable table which collects traffic counters per interface;
- o ipNetToPhysicalTable table which is the content of the Neighbor cache, i.e., the mapping between IPv6 and data-link layer addresses.

There are also YANG modules relating to the two IP addresses families and can be used with [RFC6241] and [RFC8040]. This memo recommends the use of:

- o interfaces-state/interface/statistics from ietf-interfaces@2018-02-20.yang [RFC8343] which contains counters for interfaces.
- o ipv6/neighbor from ietf-ip@2018-02-22.yang [RFC8344] which is the content of the Neighbor cache, i.e., the mapping between IPv6 and data-link layer addresses.

2.6.1.4. Neighbor Cache of IPv6 Routers

The neighbor cache of routers contains all mappings between IPv6 addresses and data-link layer addresses. There are multiple ways to collect the current entries in the Neighbor Cache, notably but not limited to:

- o the SNMP MIB (Section 2.6.1.3) as explained above;
- o using streaming telemetry or NETCONF [RFC6241] and RESTCONF [RFC8040] to collect the operational state of the neighbor cache;
- o also, by connecting over a secure management channel (such as SSH) and explicitly requesting a neighbor cache dump via the Command Line Interface (CLI) or another monitoring mechanism.

The neighbor cache is highly dynamic as mappings are added when a new IPv6 address appears on the network. This could be quite frequently with privacy extension addresses [RFC8981] or when they are removed when the state goes from UNREACH to removed (the default time for a removal per Neighbor Unreachability Detection [RFC4861] algorithm is 38 seconds for a host using Windows 7). This means that the content of the neighbor cache must periodically be fetched at an interval

which does not exhaust the router resources and still provides valuable information (suggested value is 30 seconds but this should be verified in the actual deployment) and stored for later use.

This is an important source of information because it is trivial (on a switch not using the SAVI [RFC7039] algorithm) to defeat the mapping between data-link layer address and IPv6 address. Let us rephrase the previous statement: having access to the current and past content of the neighbor cache has a paramount value for the forensic and audit trail. It should also be noted that in certain threat models this information is also deemed valuable and could itself be a target.

When using one /64 per host (Section 2.1.8) or DHCP-PD, it is sufficient to keep the history of the allocated prefixes when combined with strict source address prefix enforcement on the routers and layer-2 switches to prevent IPv6 spoofing.

2.6.1.5. Stateful DHCPv6 Lease

In some networks, IPv6 addresses/prefixes are managed by a stateful DHCPv6 server [RFC8415] that leases IPv6 addresses/prefixes to clients. It is indeed quite similar to DHCP for IPv4, so it can be tempting to use this DHCP lease file to discover the mapping between IPv6 addresses/prefixes and data-link layer addresses as is commonly used in IPv4 networking.

It is not so easy in the IPv6 networks, because not all nodes will use DHCPv6 (there are nodes which can only do stateless autoconfiguration) but also because DHCPv6 clients are identified not by their hardware-client address as in IPv4 but by a DHCP Unique ID (DUID), which can have several formats: some being the data-link layer address, some being data-link layer address prepended with time information, or even an opaque number that requires correlation with another data source to be usable for operational security. Moreover, when the DUID is based on the data-link address, this address can be of any client interface (such as the wireless interface while the client actually uses its wired interface to connect to the network).

If a lightweight DHCP relay agent [RFC6221] is used in a layer-2 switch, then the DHCP servers also receive the Interface-ID information which could be saved in order to identify the interface on which the switch received a specific leased IPv6 address. Also, if a 'normal' (not lightweight) relay agent adds the data-link layer address in the option for Relay Agent Remote-ID [RFC4649] or [RFC6939], then the DHCPv6 server can keep track of the data-link and leased IPv6 addresses.

In short, the DHCPv6 lease file is less interesting than for IPv4 networks. If possible, it is recommended to use DHCPv6 servers that keep the relayed data-link layer address in addition to the DUID in the lease file as those servers have the equivalent information to IPv4 DHCP servers.

The mapping between data-link layer address and the IPv6 address can be secured by deploying switches implementing the SAVI [RFC7513] mechanisms. Of course, this also requires that the data-link layer address is protected by using a layer-2 mechanism such as [IEEE-802.1X].

2.6.1.6. RADIUS Accounting Log

For interfaces where the user is authenticated via a RADIUS [RFC2866] server, and if RADIUS accounting is enabled, then the RADIUS server receives accounting Acct-Status-Type records at the start and at the end of the connection which include all IPv6 (and IPv4) addresses used by the user. This technique can be used notably for Wi-Fi networks with Wi-Fi Protected Address (WPA) or other IEEE 802.1X [IEEE-802.1X] wired interface on an Ethernet switch.

2.6.1.7. Other Data Sources

There are other data sources for log information that must be collected (as currently collected in IPv4 networks):

- o historical mapping of IPv6 addresses to users of remote access VPN;
- o historical mappings of MAC addresses to switch ports in a wired network.

2.6.2. Use of Collected Data

This section leverages the data collected as described before (Section 2.6.1) in order to achieve several security benefits. Section 9.1 of [RFC7934] contains more details about host tracking.

2.6.2.1. Forensic and User Accountability

The forensic use case is when the network operator must locate an IPv6 address (and the associated port, access point/switch, or VPN tunnel) that was present in the network at a certain time or is currently in the network.

To locate an IPv6 address in an enterprise network where the operator has control over all resources, the source of information can be the

neighbor cache, or, if not found, the DHCP lease file. Then, the procedure is:

1. Based on the IPv6 prefix of the IPv6 address, find the router(s) which is(are) used to reach this prefix (assuming that anti-spoofing mechanisms are used) perhaps based on an IPAM.
2. Based on this limited set of routers, on the incident time and on the IPv6 address, retrieve the data-link address from the live neighbor cache, from the historical neighbor cache data, or from SAVI events, or retrieve the data-link address from the DHCP lease file (Section 2.6.1.5).
3. Based on the data-link layer address, look-up the switch interface associated with the data-link layer address. In the case of wireless LAN with RADIUS accounting (see Section 2.6.1.6), the RADIUS log has the mapping between the user identification and the MAC address. If a Configuration Management Data Base (CMDB) is used, then it can be used to map the data-link layer address to a switch port.

At the end of the process, the interface of the host originating, or the subscriber identity associated with, the activity in question has been determined.

To identify the subscriber of an IPv6 address in a residential Internet Service Provider, the starting point is the DHCP-PD leased prefix covering the IPv6 address; this prefix can often be linked to a subscriber via the RADIUS log. Alternatively, the Forwarding Information Base (FIB) of the Cable Modem Termination System (CMTS) or Broadband Network Gateway (BNG) indicates the CPE of the subscriber and the RADIUS log can be used to retrieve the actual subscriber.

More generally, a mix of the above techniques can be used in most, if not all, networks.

2.6.2.2. Inventory

RFC 7707 [RFC7707] describes the difficulties for an attacker to scan an IPv6 network due to the vast number of IPv6 addresses per link (and why in some cases it can still be done). While the huge addressing space can sometimes be perceived as a 'protection', it also makes the inventory task difficult in an IPv6 network while it was trivial to do in an IPv4 network (a simple enumeration of all IPv4 addresses, followed by a ping and a TCP/UDP port scan). Getting an inventory of all connected devices is of prime importance for a secure network operation.

There are many ways to do an inventory of an IPv6 network.

The first technique is to use passive inspection such as IPFIX. Using exported IPFIX information and extracting the list of all IPv6 source addresses allows finding all IPv6 nodes that sent packets through a router. This is very efficient but, alas, will not discover silent nodes that never transmitted packets traversing the IPFIX target router. Also, it must be noted that link-local addresses will never be discovered by this means.

The second way is again to use the collected neighbor cache content to find all IPv6 addresses in the cache. This process will also discover all link-local addresses. See Section 2.6.1.4.

Another way that works only for a local network, consists of sending a ICMP ECHO_REQUEST to the link-local multicast address ff02::1 which addresses all IPv6 nodes on the network. All nodes should reply to this ECHO_REQUEST per [RFC4443].

Other techniques involve obtaining data from DNS, parsing log files, leveraging service discovery such as mDNS [RFC6762] and [RFC6763].

Enumerating DNS zones, especially looking at reverse DNS records and CNAMEs, is another common method employed by various tools. As already mentioned in [RFC7707], this allows an attacker to prune the IPv6 reverse DNS tree, and hence enumerate it in a feasible time. Furthermore, authoritative servers that allow zone transfers (AXFR) may be a further information source. An interesting research paper has analysed the entropy in various IPv6 addresses: see [ENTROPYIP].

2.6.2.3. Correlation

In an IPv4 network, it is easy to correlate multiple logs, for example to find events related to a specific IPv4 address. A simple Unix grep command is enough to scan through multiple text-based files and extract all lines relevant to a specific IPv4 address.

In an IPv6 network, this is slightly more difficult because different character strings can express the same IPv6 address. Therefore, the simple Unix grep command cannot be used. Moreover, an IPv6 node can have multiple IPv6 addresses.

In order to do correlation in IPv6-related logs, it is advised to have all logs in a format with only canonical IPv6 addresses [RFC5952]. Then, the neighbor cache current (or historical) data set must be searched to find the data-link layer address of the IPv6 address. Then, the current and historical neighbor cache data sets must be searched for all IPv6 addresses associated with this data-

link layer address to derive the search set. The last step is to search in all log files (containing only IPv6 addresses in canonical format) for any IPv6 addresses in the search set.

Moreover, [RFC7934] recommends using multiple IPv6 addresses per prefix, so, the correlation must also be done among those multiple IPv6 addresses, for example by discovering in the NDP cache (Section 2.6.1.4) all IPv6 addresses associated with the same MAC address and interface.

2.6.2.4. Abnormal Behavior Detection

Abnormal behavior (such as network scanning, spamming, denial-of-service) can be detected in the same way as in an IPv4 network.

- o Sudden increase of traffic detected by interface counter (SNMP) or by aggregated traffic from IPFIX records [RFC7012].
- o Rapid growth of ND cache size.
- o Change in traffic pattern (number of connections per second, number of connections per host...) observed with the use of IPFIX [RFC7012].

2.6.3. Summary

While some data sources (IPFIX, MIB, switch CAM tables, logs, ...) used in IPv4 are also used in the secure operation of an IPv6 network, the DHCPv6 lease file is less reliable and the neighbor cache is of prime importance.

The fact that there are multiple ways to express the same IPv6 address in a character string renders the use of filters mandatory when correlation must be done.

2.7. Transition/Coexistence Technologies

As it is expected that some networks will not run in a pure IPv6-only mode, the different transition mechanisms must be deployed and operated in a secure way. This section proposes operational guidelines for the most known and deployed transition techniques. [RFC4942] also contains security considerations for transition or coexistence scenarios.

2.7.1. Dual Stack

Dual stack is often the first deployment choice for network operators. Dual stacking the network offers some advantages over other transition mechanisms. Firstly, the impact on existing IPv4 operations is reduced. Secondly, in the absence of tunnels or address translation, the IPv4 and IPv6 traffic are native (easier to observe and secure) and should have the same network processing (network path, quality of service, ...). Dual stack enables a gradual termination of the IPv4 operations when the IPv6 network is ready for prime time. On the other hand, the operators have to manage two network stacks with the added complexities.

From an operational security perspective, this now means that the network operator has twice the exposure. One needs to think about protecting both protocols now. At a minimum, the IPv6 portion of a dual-stacked network should be consistent with IPv4 from a security policy point of view. Typically, the following methods are employed to protect IPv4 networks at the edge or security perimeter:

- o ACLs to permit or deny traffic;
- o Firewalls with stateful packet inspection;
- o Application firewalls inspecting the application flows.

It is recommended that these ACLs and/or firewalls be additionally configured to protect IPv6 communications. The enforced IPv6 security must be congruent with the IPv4 security policy, otherwise the attacker will use the protocol version having the more relaxed security policy. Maintaining the congruence between security policies can be challenging (especially over time); it is recommended to use a firewall or an ACL manager that is dual-stack, i.e., a system that can apply a single ACL entry to a mixed group of IPv4 and IPv6 addresses.

Application firewalls work at the application layer and are oblivious to the IP version, i.e., they work as well for IPv6 as for IPv4 and the same application security policy will work for both protocol versions.

Also, given the end-to-end connectivity that IPv6 provides, it is recommended that hosts be fortified against threats. General device hardening guidelines are provided in Section 2.8.

For many years, all host operating systems have IPv6 enabled by default, so, it is possible even in an 'IPv4-only' network to attack layer-2 adjacent victims via their IPv6 link-local address or via a

global IPv6 address when the attacker provides rogue RAs or a rogue DHCPv6 service.

[RFC7123] discusses the security implications of native IPv6 support and IPv6 transition/coexistence technologies on "IPv4-only" networks and describes possible mitigations for the aforementioned issues.

2.7.2. Encapsulation Mechanisms

There are many tunnels used for specific use cases. Except when protected by IPsec [RFC4301] or alternative tunnel encryption methods, all those tunnels have a number of security issues as described in RFC 6169 [RFC6169];

- o tunnel injection: a malevolent actor knowing a few pieces of information (for example the tunnel endpoints and the encapsulation protocol) can forge a packet which looks like a legitimate and valid encapsulated packet that will gladly be accepted by the destination tunnel endpoint. This is a specific case of spoofing;
- o traffic interception: no confidentiality is provided by the tunnel protocols (without the use of IPsec or alternative encryption methods), therefore anybody on the tunnel path can intercept the traffic and have access to the clear-text IPv6 packet; combined with the absence of authentication, an on-path attack can also be mounted;
- o service theft: as there is no authorization, even a non-authorized user can use a tunnel relay for free (this is a specific case of tunnel injection);
- o reflection attack: another specific use case of tunnel injection where the attacker injects packets with an IPv4 destination address not matching the IPv6 address causing the first tunnel endpoint to re-encapsulate the packet to the destination... Hence, the final IPv4 destination will not see the original IPv4 address but only the IPv4 address of the relay router.
- o bypassing security policy: if a firewall or an Intrusion Prevention System (IPS) is on the path of the tunnel, then it may neither inspect nor detect malevolent IPv6 traffic transmitted over the tunnel.

To mitigate the bypassing of security policies, it is often recommended to block all automatic tunnels in default OS configuration (if they are not required) by denying IPv4 packets matching:

- o IP protocol 41: this will block ISATAP (Section 2.7.2.2), 6to4 (Section 2.7.2.7), 6rd (Section 2.7.2.3), as well as, 6in4 (Section 2.7.2.1) tunnels;
- o IP protocol 47: this will block GRE (Section 2.7.2.1) tunnels;
- o UDP port 3544: this will block the default encapsulation of Teredo (Section 2.7.2.8) tunnels.

Ingress filtering [RFC2827] should also be applied on all tunnel endpoints if applicable to prevent IPv6 address spoofing.

The reflection attack cited above should also be prevented by using an IPv6 ACL preventing the hair pinning of the traffic.

As several of the tunnel techniques share the same encapsulation (i.e., IPv4 protocol 41) and embed the IPv4 address in the IPv6 address, there are a set of well-known looping attacks described in RFC 6324 [RFC6324]. This RFC also proposes mitigation techniques.

2.7.2.1. Site-to-Site Static Tunnels

Site-to-site static tunnels are described in RFC 2529 [RFC2529] and in GRE [RFC2784]. As the IPv4 endpoints are statically configured and are not dynamic, they are slightly more secure (bi-directional service theft is mostly impossible) but traffic interception and tunnel injection are still possible. Therefore, the use of IPsec [RFC4301] in transport mode to protect the encapsulated IPv4 packets is recommended for those tunnels. Alternatively, IPsec in tunnel mode can be used to transport IPv6 traffic over a non-trusted IPv4 network.

2.7.2.2. ISATAP

ISATAP tunnels [RFC5214] are mainly used within a single administrative domain and to connect a single IPv6 host to the IPv6 network. This often implies that those systems are usually managed by a single entity; therefore, audit trail and strict anti-spoofing are usually possible and this raises the overall security. Even if ISATAP is no more often used, its security issues are relevant per [KRISTOFF].

Special care must be taken to avoid a looping attack by implementing the measures of [RFC6324] and [RFC6964] (especially the section 3.6).

IPsec [RFC4301] in transport or tunnel mode can be used to secure the IPv4 ISATAP traffic to provide IPv6 traffic confidentiality and prevent service theft.

2.7.2.3. 6rd

While 6rd tunnels share the same encapsulation as 6to4 tunnels (Section 2.7.2.7), they are designed to be used within a single SP domain, in other words, they are deployed in a more constrained environment (e.g., anti-spoofing, protocol 41 filtering at the edge) than 6to4 tunnels and have few security issues other than lack of confidentiality. The security considerations (Section 12) of [RFC5969] describes how to secure 6rd tunnels.

IPsec [RFC4301] for the transported IPv6 traffic can be used if confidentiality is important.

2.7.2.4. 6PE, 6VPE, and LDPv6

Organizations using MPLS in their core can also use 6PE [RFC4798] and 6VPE [RFC4659] to enable IPv6 access over MPLS. As 6PE and 6VPE are really similar to BGP/MPLS IP VPNs described in [RFC4364], the security properties of these networks are also similar to those described in [RFC4381] (please note that this RFC may resemble a published IETF work but it is not based on an IETF review and the IETF disclaims any knowledge of the fitness of this RFC for any purpose). They rely on:

- o Address space, routing, and traffic separation with the help of VRFs (only applicable to 6VPE);
- o Hiding the IPv4 core, hence removing all attacks against P-routers;
- o Securing the routing protocol between CE and PE; in the case of 6PE and 6VPE, link-local addresses (see [RFC7404]) can be used and as these addresses cannot be reached from outside of the link, the security of 6PE and 6VPE is even higher than an IPv4 BGP/MPLS IP VPN.

LDPv6 itself does not induce new risks, see also [RFC7552].

2.7.2.5. DS-Lite

DS-lite is also a translation mechanism and is therefore analyzed further (Section 2.7.3.3) in this document as it includes IPv4 NAT.

2.7.2.6. Mapping of Address and Port

With the encapsulation and translation versions of mapping of Address and Port (MAP) (MAP-E [RFC7597] and MAP-T [RFC7599]), the access network is purely an IPv6 network and MAP protocols are used to

provide IPv4 hosts on the subscriber network access to IPv4 hosts on the Internet. The subscriber router does stateful operations in order to map all internal IPv4 addresses and layer-4 ports to the IPv4 address and the set of layer-4 ports received through the MAP configuration process. The SP equipment always does stateless operations (either decapsulation or stateless translation). Therefore, as opposed to Section 2.7.3.3, there is no state-exhaustion DoS attack against the SP equipment because there is no state and there is no operation caused by a new layer-4 connection (no logging operation).

The SP MAP equipment should implement all the security considerations of [RFC7597]; notably, ensuring that the mapping of the IPv4 address and port are consistent with the configuration. As MAP has a predictable IPv4 address and port mapping, the audit logs are easier to use as there is a clear mapping between the IPv6 address and the IPv4 address and ports.

2.7.2.7. 6to4

In [RFC3056]; 6to4 tunnels require a public routable IPv4 address in order to work correctly. They can be used to provide either single IPv6 host connectivity to the IPv6 Internet or multiple IPv6 networks connectivity to the IPv6 Internet. The 6to4 relay was historically the anycast address defined in [RFC3068] which has been deprecated by [RFC7526] and is no longer used by recent Operating Systems. Some security considerations are explained in [RFC3964].

[RFC6343] points out that if an operator provides well-managed servers and relays for 6to4, non-encapsulated IPv6 packets will pass through well-defined points (the native IPv6 interfaces of those servers and relays) at which security mechanisms may be applied. Client usage of 6to4 by default is now discouraged, and significant precautions are needed to avoid operational problems.

2.7.2.8. Teredo

Teredo tunnels [RFC4380] are mainly used in a residential environment because Teredo easily traverses an IPv4 NAT device thanks to its UDP encapsulation. Teredo tunnels connect a single host to the IPv6 Internet. Teredo shares the same issues as other tunnels: no authentication, no confidentiality, possible spoofing and reflection attacks.

IPsec [RFC4301] for the transported IPv6 traffic is recommended.

The biggest threat to Teredo is probably for an IPv4-only network as Teredo has been designed to easily traverse IPv4 NAT-PT devices which

are quite often co-located with a stateful firewall. Therefore, if the stateful IPv4 firewall allows unrestricted UDP outbound and accepts the return UDP traffic, then Teredo actually punches a hole in this firewall for all IPv6 traffic to the Internet and from the Internet. Host policies can be deployed to block Teredo in an IPv4-only network in order to avoid this firewall bypass. On the IPv4 firewall all outbound UDP should be blocked except for the commonly used services (e.g., port 53 for DNS, port 123 for NTP, port 443 for QUIC, port 500 for IKE, port 3478 for STUN, etc.).

Teredo is now hardly ever used and no longer enabled by default in most environments, so it is less of a threat, however, special consideration must be taken in cases when devices with older or non-updated operating systems may be present and by default were running Teredo.

2.7.3. Translation Mechanisms

Translation mechanisms between IPv4 and IPv6 networks are alternate coexistence strategies while networks transition to IPv6. While a framework is described in [RFC6144], the specific security considerations are documented with each individual mechanism. For the most part, they specifically mention interference with IPsec or DNSSEC deployments, how to mitigate spoofed traffic, and what some effective filtering strategies may be.

While not really a transition mechanism to IPv6, this section also includes the discussion about the use of heavy IPv4-to-IPv4 network address and port translation to prolong the life of IPv4-only networks.

2.7.3.1. Carrier-Grade NAT (CGN)

Carrier-Grade NAT (CGN), also called NAT444 CGN or Large Scale NAT (LSN) or SP NAT is described in [RFC6264] and is utilized as an interim measure to extend the use of IPv4 in a large service provider network until the provider can deploy an effective IPv6 solution. [RFC6598] requested a specific IANA allocated /10 IPv4 address block to be used as address space shared by all access networks using CGN. This has been allocated as 100.64.0.0/10.

Section 13 of [RFC6269] lists some specific security-related issues caused by large scale address sharing. The Security Considerations section of [RFC6598] also lists some specific mitigation techniques for potential misuse of shared address space. Some Law Enforcement Agencies have identified CGN as impeding their cyber-crime investigations (for example Europol press release on CGN [europol-cgn]). Many translation techniques (NAT64, DS-lite, ...)

have the same security issues as CGN when one part of the connection is IPv4-only.

[RFC6302] has recommendations for Internet-facing servers to also log the source TCP or UDP ports of incoming connections in an attempt to help identify the users behind such a CGN.

[RFC7422] suggests the use of deterministic address mapping in order to reduce logging requirements for CGN. The idea is to have a known algorithm for mapping the internal subscriber to/from public TCP and UDP ports.

[RFC6888] lists common requirements for CGNs. [RFC6967] analyzes some solutions to enforce policies on misbehaving nodes when address sharing is used. [RFC7857] also updates the NAT behavioral requirements.

2.7.3.2. NAT64/DNS64 and 464XLAT

Stateful NAT64 translation [RFC6146] allows IPv6-only clients to contact IPv4 servers using unicast UDP, TCP, or ICMP. It can be used in conjunction with DNS64 [RFC6147], a mechanism which synthesizes AAAA records from existing A records. There is also a stateless NAT64 [RFC7915], which has similar security aspects but with the added benefit of being stateless, so, less prone to a state exhaustion attack.

The Security Consideration sections of [RFC6146] and [RFC6147] list the comprehensive issues; in section 8 of [RFC6147] there are some considerations on the interaction between NAT64 and DNSSEC. A specific issue with the use of NAT64 is that it will interfere with most IPsec deployments unless UDP encapsulation is used.

Another translation mechanism relying on a combination of stateful and stateless translation, 464XLAT [RFC6877], can be used to do host local translation from IPv4 to IPv6 and a network provider translation from IPv6 to IPv4, i.e., giving IPv4-only application access to an IPv4-only server over an IPv6-only network. 464XLAT shares the same security considerations as NAT64 and DNS64, however it can be used without DNS64, avoiding the DNSSEC implications.

2.7.3.3. DS-Lite

Dual-Stack Lite (DS-Lite) [RFC6333] is a transition technique that enables a service provider to share IPv4 addresses among customers by combining two well-known technologies: IP in IP (IPv4-in-IPv6) and IPv4 NAPT.

Security considerations with respect to DS-Lite mainly revolve around logging data, preventing DoS attacks from rogue devices (as the Address Family Translation Router (AFTR) [RFC6333] function is stateful) and restricting service offered by the AFTR only to registered customers.

Section 11 of [RFC6333] and section 2 of [RFC7785] describe important security issues associated with this technology.

2.8. General Device Hardening

With almost all devices being IPv6 enabled by default and with many end points having IPv6 connectivity to the Internet, it is critical to also harden those devices against attacks over IPv6.

The same techniques used to protect devices against attack over IPv4 should be used for IPv6 and should include, but not limited to:

- o Restrict device access to authorized individuals
- o Monitor and audit access to the device
- o Turn off any unused services on the end node
- o Understand which IPv6 addresses are being used to source traffic and change defaults if necessary
- o Use cryptographically protected protocols for device management (SCP, SNMPv3, SSH, TLS, etc.)
- o Use host firewall capabilities to control traffic that gets processed by upper-layer protocols
- o apply firmware, OS and application patches/upgrades to the devices in a timely manner
- o use multi-factor credentials to authenticate to devices
- o Use virus scanners to detect malicious programs

3. Enterprises Specific Security Considerations

Enterprises [RFC7381] generally have robust network security policies in place to protect existing IPv4 networks. These policies have been distilled from years of experiential knowledge of securing IPv4 networks. At the very least, it is recommended that enterprise networks have parity between their security policies for both protocol versions. This section also applies to the enterprise part

of all SP networks, i.e., the part of the network where the SP employees are connected.

Security considerations in the enterprise can be broadly categorized into two groups: External and Internal.

3.1. External Security Considerations

The external aspect deals with providing security at the edge or perimeter of the enterprise network where it meets the service provider's network. This is commonly achieved by enforcing a security policy either by implementing dedicated firewalls with stateful packet inspection or a router with ACLs. A common default IPv4 policy on firewalls that could easily be ported to IPv6 is to allow all traffic outbound while only allowing specific traffic, such as established sessions, inbound (see also [RFC6092]). Section 3.2 of [RFC7381] also provides similar recommendations.

Here are a few more things that could enhance the default policy:

- o Filter internal-use IPv6 addresses at the perimeter, this will also mitigate the vulnerabilities listed in [RFC7359]
- o Discard packets from and to bogon and reserved space, see also [CYMRU] and [RFC8190]
- o Accept certain ICMPv6 messages to allow proper operation of ND and PMTUD, see also [RFC4890] or [REY_PF] for hosts
- o Based on the use of the network, filter specific extension headers by accepting only the required ones (permit list approach) such as ESP, AH, and not forgetting the required transport layers: ICMP, TCP, UDP, ... This filtering should be done where applicable at the edge and possibly inside the perimeter; see also [I-D.ietf-opsec-ipv6-eh-filtering]
- o Filter packets having an illegal IPv6 headers chain at the perimeter (and if possible, inside the network as well), see Section 2.2
- o Filter unneeded services at the perimeter
- o Implement ingress and egress anti-spoofing in the forwarding and control planes, see [RFC2827] and [RFC3704]
- o Implement appropriate rate-limiters and control-plane policers based on traffic baselines

Having global IPv6 addresses on all the enterprise sites is different than in IPv4 where [RFC1918] addresses are often used internally and not routed over the Internet. [RFC7359] and [WEBER_VPN] explain that without careful design, there could be IPv6 leakages from layer-3 VPNs.

3.2. Internal Security Considerations

The internal aspect deals with providing security inside the perimeter of the network, including end hosts. Internal networks of enterprises are often different: University campus, wireless guest access, ... so there is no "one size fits all" recommendation.

The most significant concerns here are related to Neighbor Discovery. At the network level, it is recommended that all security considerations discussed in Section 2.3 be reviewed carefully and the recommendations be considered in-depth as well. Section 4.1 of [RFC7381] also provides some recommendations.

As mentioned in Section 2.7.2, care must be taken when running automated IPv6-in-IPv4 tunnels.

When site-to-site VPNs are used it should be kept in mind that, given the global scope of IPv6 global addresses as opposed to the common use of IPv4 private address space [RFC1918], sites might be able to communicate with each other over the Internet even when the VPN mechanism is not available and hence no traffic encryption is performed and traffic could be injected from the Internet into the site, see [WEBER_VPN]. It is recommended to filter at Internet connection(s) packets having a source or destination address belonging to the site internal prefix(es); this should be done for ingress and egress traffic.

Hosts need to be hardened directly through security policy to protect against security threats. The host firewall default capabilities have to be clearly understood. In some cases, 3rd party firewalls have no IPv6 support whereas the native firewall installed by default has IPv6 support. General device hardening guidelines are provided in Section 2.8.

It should also be noted that many hosts still use IPv4 for transporting logs for RADIUS, DIAMETER, TACACS+, SYSLOG, etc. Operators cannot rely on an IPv6-only security policy to secure such protocols that are still using IPv4.

4. Service Providers Security Considerations

4.1. BGP

The threats and mitigation techniques are identical between IPv4 and IPv6. Broadly speaking they are:

- o Authenticating the TCP session;
- o TTL security (which becomes hop-limit security in IPv6) as [RFC5082];
- o bogon AS filtering, see [CYMRU];
- o Prefix filtering.

These are explained in more detail in Section 2.5. Also, the recommendations of [RFC7454] should be considered.

4.1.1. Remote Triggered Black Hole Filtering (RTBH)

RTBH [RFC5635] works identically in IPv4 and IPv6. IANA has allocated the 100::/64 prefix to be used as the discard prefix [RFC6666]

4.2. Transition/Coexistence Mechanism

SPs will typically use transition mechanisms such as 6rd, 6PE, MAP, and NAT64 which have been analyzed in the transition and coexistence Section 2.7 section.

4.3. Lawful Intercept

The Lawful Intercept requirements are similar for IPv6 and IPv4 architectures and will be subject to the laws enforced in different geographic regions. The local issues with each jurisdiction can make this challenging and both corporate legal and privacy personnel should be involved in discussions pertaining to what information gets logged and with regard to the respective log retention policies for this information.

The target of interception will usually be a residential subscriber (e.g., his/her PPP session, physical line, or CPE MAC address). In the absence of IPv6 NAT on the CPE, IPv6 has the possibility to allow for intercepting the traffic from a single host (i.e., a /128 target) rather than the whole set of hosts of a subscriber (which could be a /48, /60, or /64).

In contrast, in mobile environments, since the 3GPP specifications allocate a /64 per device, it may be sufficient to intercept traffic from the /64 rather than specific /128's (since each time the device establishes a data connection it gets a new IID).

5. Residential Users Security Considerations

The IETF Homenet working group is working on standards and guidelines for IPv6 residential networks; this obviously includes operational security considerations; but this is still work in progress. [RFC8520] is an interesting approach on how firewalls could retrieve and apply specific security policies to some residential devices.

Some residential users have less experience and knowledge about security or networking than experimented operators. As most of the recent hosts (e.g., smartphones, tablets) have IPv6 enabled by default, IPv6 security is important for those users. Even with an IPv4-only ISP, those users can get IPv6 Internet access with the help of Teredo (Section 2.7.2.8) tunnels. Several peer-to-peer programs support IPv6 and those programs can initiate a Teredo tunnel through an IPv4 residential gateway, with the consequence of making the internal host reachable from any IPv6 host on the Internet. It is therefore recommended that all host security products (including personal firewalls) are configured with a dual-stack security policy.

If the residential CPE has IPv6 connectivity, [RFC7084] defines the requirements of an IPv6 CPE and does not take a position on the debate of default IPv6 security policy as defined in [RFC6092]:

- o outbound only: allowing all internally initiated connections and block all externally initiated ones, which is a common default security policy enforced by IPv4 Residential Gateway doing NAT but it also breaks the end-to-end reachability promise of IPv6. [RFC6092] lists several recommendations to design such a CPE;
- o open/transparent: allowing all internally and externally initiated connections, therefore restoring the end-to-end nature of the Internet for IPv6 traffic but having a different security policy for IPv6 than for IPv4.

[RFC6092] REC-49 states that a choice must be given to the user to select one of those two policies.

6. Further Reading

There are several documents that describe in more detail the security of an IPv6 network; these documents are not written by the IETF and

some of them are dated but are listed here for the reader's convenience:

1. Guidelines for the Secure Deployment of IPv6 [NIST]
2. North American IPv6 Task Force Technology Report - IPv6 Security Technology Paper [NAv6TF_Security]
3. IPv6 Security [IPv6_Security_Book]

7. Acknowledgements

The authors would like to thank the following people for their useful comments: Mikael Abrahamsson, Fred Baker, Mustafa Suha Botsali, Mohamed Boucadair, Brian Carpenter, Tim Chown, Lorenzo Colitti, Roman Danyliw (IESG review), Markus de Bruen, Lars Eggert (IESG review), Tobias Fiebig, Fernando Gont, Jeffry Handal, Lee Howard, Benjamin Kaduk (IESG review), Panos Kampanakis, Erik Kline, Jouni Korhonen, Warren Kumari (IESG review), Ted Lemon, Mark Lentczner, Acee Lindem (and his detailed nits), Jen Linkova (and her detailed review), Gyan S. Mishra (the document shepherd), Jordi Palet, Alvaro Retana (IESG review), Zaheduzzaman Sarker (IESG review), Bob Sleigh, Donald Smith, Tarko Tikan, Ole Troan, Bernie Volz (by alphabetical order).

8. Security Considerations

This memo attempts to give an overview of security considerations of operating an IPv6 network both for an IPv6-only network and for networks utilizing the most widely deployed IPv4/IPv6 coexistence strategies.

9. References

9.1. Normative References

- [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>>.

9.2. Informative References

- [CYMRU] Team, C., "The Bogon Reference", Existing in 2021, <<https://team-cymru.com/community-services/bogon-reference/>>.

[ENTROPYIP]

Foremski, P., Plonka, D., and A. Berger, "Entropy/IP: Uncovering Structure in IPv6 Addresses",
<<http://www.entropy-ip.com/>>.

[europol-cgn]

Europol, "ARE YOU SHARING THE SAME IP ADDRESS AS A CRIMINAL? LAW ENFORCEMENT CALL FOR THE END OF CARRIER GRADE NAT (CGN) TO INCREASE ACCOUNTABILITY ONLINE", October 2017,
<<https://www.europol.europa.eu/newsroom/news/are-you-sharing-same-ip-address-criminal-law-enforcement-call-for-end-of-carrier-grade-nat-cgn-to-increase-accountability-online>>.

[GDPR]

Union, O. J. O. T. E., "Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)", April 2016,
<<https://eur-lex.europa.eu/eli/reg/2016/679/oj>>.

[I-D.ietf-opsec-ipv6-eh-filtering]

Gont, F. and W. Liu, "Recommendations on the Filtering of IPv6 Packets Containing IPv6 Extension Headers at Transit Routers", draft-ietf-opsec-ipv6-eh-filtering-07 (work in progress), January 2021.

[I-D.kampanakis-6man-ipv6-eh-parsing]

Kampanakis, P., "Implementation Guidelines for parsing IPv6 Extension Headers", draft-kampanakis-6man-ipv6-eh-parsing-01 (work in progress), August 2014.

[IANA-IPFIX]

IANA, "IP Flow Information Export (IPFIX) Entities",
<<http://www.iana.org/assignments/ipfix>>.

[IEEE-802.1X]

IEEE, "IEEE Standard for Local and metropolitan area networks - Port-Based Network Access Control", IEEE Std 802.1X-2010, February 2010.

[IPv6_Security_Book]

Hogg, S. and E. Vyncke, "IPv6 Security",
ISBN 1-58705-594-5, Publisher CiscoPress, December 2008.

[KRISTOFF]

Kristoff, J., Ghasemisharif, M., Kanich, C., and J. Polakis, "Plight at the End of the Tunnel: Legacy IPv6 Transition Mechanisms in the Wild", March 2021, <<https://dataplane.org/jtk/publications/kgkp-pam-21.pdf>>.

[NAv6TF_Security]

Kaeo, M., Green, D., Bound, J., and Y. Pouffary, "North American IPv6 Task Force Technology Report - IPv6 Security Technology Paper", 2006, <http://www.ipv6forum.com/dl/white/NAv6TF_Security_Report.pdf>.

[NIST]

Frankel, S., Graveman, R., Pearce, J., and M. Rocks, "Guidelines for the Secure Deployment of IPv6", 2010, <<http://csrc.nist.gov/publications/nistpubs/800-119/sp800-119.pdf>>.

[RADB]

INC., M. N., "RADb The Internet Routing Registry", Existing in 2021, <<https://www.radb.net/>>.

[REY_PF]

Rey, E., "Local Packet Filtering with IPv6", July 2017, <https://labs.ripe.net/Members/enno_rey/local-packet-filtering-with-ipv6>.

[RFC0826]

Plummer, D., "An Ethernet Address Resolution Protocol: Or Converting Network Protocol Addresses to 48.bit Ethernet Address for Transmission on Ethernet Hardware", STD 37, RFC 826, DOI 10.17487/RFC0826, November 1982, <<https://www.rfc-editor.org/info/rfc826>>.

[RFC1918]

Rekhter, Y., Moskowitz, B., Karrenberg, D., de Groot, G., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, DOI 10.17487/RFC1918, February 1996, <<https://www.rfc-editor.org/info/rfc1918>>.

[RFC2131]

Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, DOI 10.17487/RFC2131, March 1997, <<https://www.rfc-editor.org/info/rfc2131>>.

[RFC2460]

Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, DOI 10.17487/RFC2460, December 1998, <<https://www.rfc-editor.org/info/rfc2460>>.

[RFC2529]

Carpenter, B. and C. Jung, "Transmission of IPv6 over IPv4 Domains without Explicit Tunnels", RFC 2529, DOI 10.17487/RFC2529, March 1999, <<https://www.rfc-editor.org/info/rfc2529>>.

- [RFC2663] Srisuresh, P. and M. Holdrege, "IP Network Address Translator (NAT) Terminology and Considerations", RFC 2663, DOI 10.17487/RFC2663, August 1999, <<https://www.rfc-editor.org/info/rfc2663>>.
- [RFC2784] Farinacci, D., Li, T., Hanks, S., Meyer, D., and P. Traina, "Generic Routing Encapsulation (GRE)", RFC 2784, DOI 10.17487/RFC2784, March 2000, <<https://www.rfc-editor.org/info/rfc2784>>.
- [RFC2827] Ferguson, P. and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", BCP 38, RFC 2827, DOI 10.17487/RFC2827, May 2000, <<https://www.rfc-editor.org/info/rfc2827>>.
- [RFC2866] Rigney, C., "RADIUS Accounting", RFC 2866, DOI 10.17487/RFC2866, June 2000, <<https://www.rfc-editor.org/info/rfc2866>>.
- [RFC3056] Carpenter, B. and K. Moore, "Connection of IPv6 Domains via IPv4 Clouds", RFC 3056, DOI 10.17487/RFC3056, February 2001, <<https://www.rfc-editor.org/info/rfc3056>>.
- [RFC3068] Huitema, C., "An Anycast Prefix for 6to4 Relay Routers", RFC 3068, DOI 10.17487/RFC3068, June 2001, <<https://www.rfc-editor.org/info/rfc3068>>.
- [RFC3627] Savola, P., "Use of /127 Prefix Length Between Routers Considered Harmful", RFC 3627, DOI 10.17487/RFC3627, September 2003, <<https://www.rfc-editor.org/info/rfc3627>>.
- [RFC3704] Baker, F. and P. Savola, "Ingress Filtering for Multihomed Networks", BCP 84, RFC 3704, DOI 10.17487/RFC3704, March 2004, <<https://www.rfc-editor.org/info/rfc3704>>.
- [RFC3756] Nikander, P., Ed., Kempf, J., and E. Nordmark, "IPv6 Neighbor Discovery (ND) Trust Models and Threats", RFC 3756, DOI 10.17487/RFC3756, May 2004, <<https://www.rfc-editor.org/info/rfc3756>>.
- [RFC3924] Baker, F., Foster, B., and C. Sharp, "Cisco Architecture for Lawful Intercept in IP Networks", RFC 3924, DOI 10.17487/RFC3924, October 2004, <<https://www.rfc-editor.org/info/rfc3924>>.
- [RFC3964] Savola, P. and C. Patel, "Security Considerations for 6to4", RFC 3964, DOI 10.17487/RFC3964, December 2004, <<https://www.rfc-editor.org/info/rfc3964>>.

- [RFC3971] Arkko, J., Ed., Kempf, J., Zill, B., and P. Nikander, "SEcure Neighbor Discovery (SEND)", RFC 3971, DOI 10.17487/RFC3971, March 2005, <<https://www.rfc-editor.org/info/rfc3971>>.
- [RFC3972] Aura, T., "Cryptographically Generated Addresses (CGA)", RFC 3972, DOI 10.17487/RFC3972, March 2005, <<https://www.rfc-editor.org/info/rfc3972>>.
- [RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, DOI 10.17487/RFC4033, March 2005, <<https://www.rfc-editor.org/info/rfc4033>>.
- [RFC4107] Bellovin, S. and R. Housley, "Guidelines for Cryptographic Key Management", BCP 107, RFC 4107, DOI 10.17487/RFC4107, June 2005, <<https://www.rfc-editor.org/info/rfc4107>>.
- [RFC4193] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast Addresses", RFC 4193, DOI 10.17487/RFC4193, October 2005, <<https://www.rfc-editor.org/info/rfc4193>>.
- [RFC4293] Routhier, S., Ed., "Management Information Base for the Internet Protocol (IP)", RFC 4293, DOI 10.17487/RFC4293, April 2006, <<https://www.rfc-editor.org/info/rfc4293>>.
- [RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005, <<https://www.rfc-editor.org/info/rfc4301>>.
- [RFC4302] Kent, S., "IP Authentication Header", RFC 4302, DOI 10.17487/RFC4302, December 2005, <<https://www.rfc-editor.org/info/rfc4302>>.
- [RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC 4303, DOI 10.17487/RFC4303, December 2005, <<https://www.rfc-editor.org/info/rfc4303>>.
- [RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, DOI 10.17487/RFC4364, February 2006, <<https://www.rfc-editor.org/info/rfc4364>>.
- [RFC4380] Huitema, C., "Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs)", RFC 4380, DOI 10.17487/RFC4380, February 2006, <<https://www.rfc-editor.org/info/rfc4380>>.

- [RFC4381] Behringer, M., "Analysis of the Security of BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4381, DOI 10.17487/RFC4381, February 2006, <<https://www.rfc-editor.org/info/rfc4381>>.
- [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>>.
- [RFC4552] Gupta, M. and N. Melam, "Authentication/Confidentiality for OSPFv3", RFC 4552, DOI 10.17487/RFC4552, June 2006, <<https://www.rfc-editor.org/info/rfc4552>>.
- [RFC4649] Volz, B., "Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Relay Agent Remote-ID Option", RFC 4649, DOI 10.17487/RFC4649, August 2006, <<https://www.rfc-editor.org/info/rfc4649>>.
- [RFC4659] De Clercq, J., Ooms, D., Carugi, M., and F. Le Faucheur, "BGP-MPLS IP Virtual Private Network (VPN) Extension for IPv6 VPN", RFC 4659, DOI 10.17487/RFC4659, September 2006, <<https://www.rfc-editor.org/info/rfc4659>>.
- [RFC4795] Aboba, B., Thaler, D., and L. Esibov, "Link-local Multicast Name Resolution (LLMNR)", RFC 4795, DOI 10.17487/RFC4795, January 2007, <<https://www.rfc-editor.org/info/rfc4795>>.
- [RFC4798] De Clercq, J., Ooms, D., Prevost, S., and F. Le Faucheur, "Connecting IPv6 Islands over IPv4 MPLS Using IPv6 Provider Edge Routers (6PE)", RFC 4798, DOI 10.17487/RFC4798, February 2007, <<https://www.rfc-editor.org/info/rfc4798>>.
- [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>>.
- [RFC4864] Van de Velde, G., Hain, T., Droms, R., Carpenter, B., and E. Klein, "Local Network Protection for IPv6", RFC 4864, DOI 10.17487/RFC4864, May 2007, <<https://www.rfc-editor.org/info/rfc4864>>.

- [RFC4890] Davies, E. and J. Mohacsi, "Recommendations for Filtering ICMPv6 Messages in Firewalls", RFC 4890, DOI 10.17487/RFC4890, May 2007, <<https://www.rfc-editor.org/info/rfc4890>>.
- [RFC4942] Davies, E., Krishnan, S., and P. Savola, "IPv6 Transition/Co-existence Security Considerations", RFC 4942, DOI 10.17487/RFC4942, September 2007, <<https://www.rfc-editor.org/info/rfc4942>>.
- [RFC5082] Gill, V., Heasley, J., Meyer, D., Savola, P., Ed., and C. Pignataro, "The Generalized TTL Security Mechanism (GTSM)", RFC 5082, DOI 10.17487/RFC5082, October 2007, <<https://www.rfc-editor.org/info/rfc5082>>.
- [RFC5214] Templin, F., Gleeson, T., and D. Thaler, "Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)", RFC 5214, DOI 10.17487/RFC5214, March 2008, <<https://www.rfc-editor.org/info/rfc5214>>.
- [RFC5340] Coltun, R., Ferguson, D., Moy, J., and A. Lindem, "OSPF for IPv6", RFC 5340, DOI 10.17487/RFC5340, July 2008, <<https://www.rfc-editor.org/info/rfc5340>>.
- [RFC5635] Kumari, W. and D. McPherson, "Remote Triggered Black Hole Filtering with Unicast Reverse Path Forwarding (uRPF)", RFC 5635, DOI 10.17487/RFC5635, August 2009, <<https://www.rfc-editor.org/info/rfc5635>>.
- [RFC5952] Kawamura, S. and M. Kawashima, "A Recommendation for IPv6 Address Text Representation", RFC 5952, DOI 10.17487/RFC5952, August 2010, <<https://www.rfc-editor.org/info/rfc5952>>.
- [RFC5969] Townsley, W. and O. Troan, "IPv6 Rapid Deployment on IPv4 Infrastructures (6rd) -- Protocol Specification", RFC 5969, DOI 10.17487/RFC5969, August 2010, <<https://www.rfc-editor.org/info/rfc5969>>.
- [RFC6092] Woodyatt, J., Ed., "Recommended Simple Security Capabilities in Customer Premises Equipment (CPE) for Providing Residential IPv6 Internet Service", RFC 6092, DOI 10.17487/RFC6092, January 2011, <<https://www.rfc-editor.org/info/rfc6092>>.
- [RFC6104] Chown, T. and S. Venaas, "Rogue IPv6 Router Advertisement Problem Statement", RFC 6104, DOI 10.17487/RFC6104, February 2011, <<https://www.rfc-editor.org/info/rfc6104>>.

- [RFC6105] Levy-Abegnoli, E., Van de Velde, G., Popoviciu, C., and J. Mohacsi, "IPv6 Router Advertisement Guard", RFC 6105, DOI 10.17487/RFC6105, February 2011, <<https://www.rfc-editor.org/info/rfc6105>>.
- [RFC6144] Baker, F., Li, X., Bao, C., and K. Yin, "Framework for IPv4/IPv6 Translation", RFC 6144, DOI 10.17487/RFC6144, April 2011, <<https://www.rfc-editor.org/info/rfc6144>>.
- [RFC6146] Bagnulo, M., Matthews, P., and I. van Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", RFC 6146, DOI 10.17487/RFC6146, April 2011, <<https://www.rfc-editor.org/info/rfc6146>>.
- [RFC6147] Bagnulo, M., Sullivan, A., Matthews, P., and I. van Beijnum, "DNS64: DNS Extensions for Network Address Translation from IPv6 Clients to IPv4 Servers", RFC 6147, DOI 10.17487/RFC6147, April 2011, <<https://www.rfc-editor.org/info/rfc6147>>.
- [RFC6164] Kohno, M., Nitzan, B., Bush, R., Matsuzaki, Y., Colitti, L., and T. Narten, "Using 127-Bit IPv6 Prefixes on Inter-Router Links", RFC 6164, DOI 10.17487/RFC6164, April 2011, <<https://www.rfc-editor.org/info/rfc6164>>.
- [RFC6169] Krishnan, S., Thaler, D., and J. Hoagland, "Security Concerns with IP Tunneling", RFC 6169, DOI 10.17487/RFC6169, April 2011, <<https://www.rfc-editor.org/info/rfc6169>>.
- [RFC6177] Narten, T., Huston, G., and L. Roberts, "IPv6 Address Assignment to End Sites", BCP 157, RFC 6177, DOI 10.17487/RFC6177, March 2011, <<https://www.rfc-editor.org/info/rfc6177>>.
- [RFC6192] Dugal, D., Pignataro, C., and R. Dunn, "Protecting the Router Control Plane", RFC 6192, DOI 10.17487/RFC6192, March 2011, <<https://www.rfc-editor.org/info/rfc6192>>.
- [RFC6221] Miles, D., Ed., Ooghe, S., Dec, W., Krishnan, S., and A. Kavanagh, "Lightweight DHCPv6 Relay Agent", RFC 6221, DOI 10.17487/RFC6221, May 2011, <<https://www.rfc-editor.org/info/rfc6221>>.
- [RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <<https://www.rfc-editor.org/info/rfc6241>>.

- [RFC6264] Jiang, S., Guo, D., and B. Carpenter, "An Incremental Carrier-Grade NAT (CGN) for IPv6 Transition", RFC 6264, DOI 10.17487/RFC6264, June 2011, <<https://www.rfc-editor.org/info/rfc6264>>.
- [RFC6269] Ford, M., Ed., Boucadair, M., Durand, A., Levis, P., and P. Roberts, "Issues with IP Address Sharing", RFC 6269, DOI 10.17487/RFC6269, June 2011, <<https://www.rfc-editor.org/info/rfc6269>>.
- [RFC6296] Wasserman, M. and F. Baker, "IPv6-to-IPv6 Network Prefix Translation", RFC 6296, DOI 10.17487/RFC6296, June 2011, <<https://www.rfc-editor.org/info/rfc6296>>.
- [RFC6302] Durand, A., Gashinsky, I., Lee, D., and S. Sheppard, "Logging Recommendations for Internet-Facing Servers", BCP 162, RFC 6302, DOI 10.17487/RFC6302, June 2011, <<https://www.rfc-editor.org/info/rfc6302>>.
- [RFC6324] Nakibly, G. and F. Templin, "Routing Loop Attack Using IPv6 Automatic Tunnels: Problem Statement and Proposed Mitigations", RFC 6324, DOI 10.17487/RFC6324, August 2011, <<https://www.rfc-editor.org/info/rfc6324>>.
- [RFC6333] Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", RFC 6333, DOI 10.17487/RFC6333, August 2011, <<https://www.rfc-editor.org/info/rfc6333>>.
- [RFC6343] Carpenter, B., "Advisory Guidelines for 6to4 Deployment", RFC 6343, DOI 10.17487/RFC6343, August 2011, <<https://www.rfc-editor.org/info/rfc6343>>.
- [RFC6434] Jankiewicz, E., Loughney, J., and T. Narten, "IPv6 Node Requirements", RFC 6434, DOI 10.17487/RFC6434, December 2011, <<https://www.rfc-editor.org/info/rfc6434>>.
- [RFC6459] Korhonen, J., Ed., Soininen, J., Patil, B., Savolainen, T., Bajko, G., and K. Iisakkila, "IPv6 in 3rd Generation Partnership Project (3GPP) Evolved Packet System (EPS)", RFC 6459, DOI 10.17487/RFC6459, January 2012, <<https://www.rfc-editor.org/info/rfc6459>>.
- [RFC6547] George, W., "RFC 3627 to Historic Status", RFC 6547, DOI 10.17487/RFC6547, February 2012, <<https://www.rfc-editor.org/info/rfc6547>>.

- [RFC6564] Krishnan, S., Woodyatt, J., Kline, E., Hoagland, J., and M. Bhatia, "A Uniform Format for IPv6 Extension Headers", RFC 6564, DOI 10.17487/RFC6564, April 2012, <<https://www.rfc-editor.org/info/rfc6564>>.
- [RFC6583] Gashinsky, I., Jaeggli, J., and W. Kumari, "Operational Neighbor Discovery Problems", RFC 6583, DOI 10.17487/RFC6583, March 2012, <<https://www.rfc-editor.org/info/rfc6583>>.
- [RFC6598] Weil, J., Kuarsingh, V., Donley, C., Liljenstolpe, C., and M. Azinger, "IANA-Reserved IPv4 Prefix for Shared Address Space", BCP 153, RFC 6598, DOI 10.17487/RFC6598, April 2012, <<https://www.rfc-editor.org/info/rfc6598>>.
- [RFC6620] Nordmark, E., Bagnulo, M., and E. Levy-Abegnoli, "FCFS SAVI: First-Come, First-Served Source Address Validation Improvement for Locally Assigned IPv6 Addresses", RFC 6620, DOI 10.17487/RFC6620, May 2012, <<https://www.rfc-editor.org/info/rfc6620>>.
- [RFC6666] Hilliard, N. and D. Freedman, "A Discard Prefix for IPv6", RFC 6666, DOI 10.17487/RFC6666, August 2012, <<https://www.rfc-editor.org/info/rfc6666>>.
- [RFC6762] Cheshire, S. and M. Krochmal, "Multicast DNS", RFC 6762, DOI 10.17487/RFC6762, February 2013, <<https://www.rfc-editor.org/info/rfc6762>>.
- [RFC6763] Cheshire, S. and M. Krochmal, "DNS-Based Service Discovery", RFC 6763, DOI 10.17487/RFC6763, February 2013, <<https://www.rfc-editor.org/info/rfc6763>>.
- [RFC6775] Shelby, Z., Ed., Chakrabarti, S., Nordmark, E., and C. Bormann, "Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)", RFC 6775, DOI 10.17487/RFC6775, November 2012, <<https://www.rfc-editor.org/info/rfc6775>>.
- [RFC6877] Mawatari, M., Kawashima, M., and C. Byrne, "464XLAT: Combination of Stateful and Stateless Translation", RFC 6877, DOI 10.17487/RFC6877, April 2013, <<https://www.rfc-editor.org/info/rfc6877>>.
- [RFC6888] Perreault, S., Ed., Yamagata, I., Miyakawa, S., Nakagawa, A., and H. Ashida, "Common Requirements for Carrier-Grade NATs (CGNs)", BCP 127, RFC 6888, DOI 10.17487/RFC6888, April 2013, <<https://www.rfc-editor.org/info/rfc6888>>.

- [RFC6939] Halwasia, G., Bhandari, S., and W. Dec, "Client Link-Layer Address Option in DHCPv6", RFC 6939, DOI 10.17487/RFC6939, May 2013, <<https://www.rfc-editor.org/info/rfc6939>>.
- [RFC6964] Templin, F., "Operational Guidance for IPv6 Deployment in IPv4 Sites Using the Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)", RFC 6964, DOI 10.17487/RFC6964, May 2013, <<https://www.rfc-editor.org/info/rfc6964>>.
- [RFC6967] Boucadair, M., Touch, J., Levis, P., and R. Penno, "Analysis of Potential Solutions for Revealing a Host Identifier (HOST_ID) in Shared Address Deployments", RFC 6967, DOI 10.17487/RFC6967, June 2013, <<https://www.rfc-editor.org/info/rfc6967>>.
- [RFC6980] Gont, F., "Security Implications of IPv6 Fragmentation with IPv6 Neighbor Discovery", RFC 6980, DOI 10.17487/RFC6980, August 2013, <<https://www.rfc-editor.org/info/rfc6980>>.
- [RFC7010] Liu, B., Jiang, S., Carpenter, B., Venaas, S., and W. George, "IPv6 Site Renumbering Gap Analysis", RFC 7010, DOI 10.17487/RFC7010, September 2013, <<https://www.rfc-editor.org/info/rfc7010>>.
- [RFC7011] Claise, B., Ed., Trammell, B., Ed., and P. Aitken, "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information", STD 77, RFC 7011, DOI 10.17487/RFC7011, September 2013, <<https://www.rfc-editor.org/info/rfc7011>>.
- [RFC7012] Claise, B., Ed. and B. Trammell, Ed., "Information Model for IP Flow Information Export (IPFIX)", RFC 7012, DOI 10.17487/RFC7012, September 2013, <<https://www.rfc-editor.org/info/rfc7012>>.
- [RFC7039] Wu, J., Bi, J., Bagnulo, M., Baker, F., and C. Vogt, Ed., "Source Address Validation Improvement (SAVI) Framework", RFC 7039, DOI 10.17487/RFC7039, October 2013, <<https://www.rfc-editor.org/info/rfc7039>>.
- [RFC7045] Carpenter, B. and S. Jiang, "Transmission and Processing of IPv6 Extension Headers", RFC 7045, DOI 10.17487/RFC7045, December 2013, <<https://www.rfc-editor.org/info/rfc7045>>.

- [RFC7050] Savolainen, T., Korhonen, J., and D. Wing, "Discovery of the IPv6 Prefix Used for IPv6 Address Synthesis", RFC 7050, DOI 10.17487/RFC7050, November 2013, <<https://www.rfc-editor.org/info/rfc7050>>.
- [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>>.
- [RFC7112] Gont, F., Manral, V., and R. Bonica, "Implications of Oversized IPv6 Header Chains", RFC 7112, DOI 10.17487/RFC7112, January 2014, <<https://www.rfc-editor.org/info/rfc7112>>.
- [RFC7113] Gont, F., "Implementation Advice for IPv6 Router Advertisement Guard (RA-Guard)", RFC 7113, DOI 10.17487/RFC7113, February 2014, <<https://www.rfc-editor.org/info/rfc7113>>.
- [RFC7123] Gont, F. and W. Liu, "Security Implications of IPv6 on IPv4 Networks", RFC 7123, DOI 10.17487/RFC7123, February 2014, <<https://www.rfc-editor.org/info/rfc7123>>.
- [RFC7166] Bhatia, M., Manral, V., and A. Lindem, "Supporting Authentication Trailer for OSPFv3", RFC 7166, DOI 10.17487/RFC7166, March 2014, <<https://www.rfc-editor.org/info/rfc7166>>.
- [RFC7217] Gont, F., "A Method for Generating Semantically Opaque Interface Identifiers with IPv6 Stateless Address Autoconfiguration (SLAAC)", RFC 7217, DOI 10.17487/RFC7217, April 2014, <<https://www.rfc-editor.org/info/rfc7217>>.
- [RFC7359] Gont, F., "Layer 3 Virtual Private Network (VPN) Tunnel Traffic Leakages in Dual-Stack Hosts/Networks", RFC 7359, DOI 10.17487/RFC7359, August 2014, <<https://www.rfc-editor.org/info/rfc7359>>.
- [RFC7381] Chittimaneni, K., Chown, T., Howard, L., Kuarsingh, V., Pouffary, Y., and E. Vyncke, "Enterprise IPv6 Deployment Guidelines", RFC 7381, DOI 10.17487/RFC7381, October 2014, <<https://www.rfc-editor.org/info/rfc7381>>.

- [RFC7404] Behringer, M. and E. Vyncke, "Using Only Link-Local Addressing inside an IPv6 Network", RFC 7404, DOI 10.17487/RFC7404, November 2014, <<https://www.rfc-editor.org/info/rfc7404>>.
- [RFC7422] Donley, C., Grundemann, C., Sarawat, V., Sundaresan, K., and O. Vautrin, "Deterministic Address Mapping to Reduce Logging in Carrier-Grade NAT Deployments", RFC 7422, DOI 10.17487/RFC7422, December 2014, <<https://www.rfc-editor.org/info/rfc7422>>.
- [RFC7454] Durand, J., Pepelnjak, I., and G. Doering, "BGP Operations and Security", BCP 194, RFC 7454, DOI 10.17487/RFC7454, February 2015, <<https://www.rfc-editor.org/info/rfc7454>>.
- [RFC7513] Bi, J., Wu, J., Yao, G., and F. Baker, "Source Address Validation Improvement (SAVI) Solution for DHCP", RFC 7513, DOI 10.17487/RFC7513, May 2015, <<https://www.rfc-editor.org/info/rfc7513>>.
- [RFC7526] Troan, O. and B. Carpenter, Ed., "Deprecating the Anycast Prefix for 6to4 Relay Routers", BCP 196, RFC 7526, DOI 10.17487/RFC7526, May 2015, <<https://www.rfc-editor.org/info/rfc7526>>.
- [RFC7552] Asati, R., Pignataro, C., Raza, K., Manral, V., and R. Papneja, "Updates to LDP for IPv6", RFC 7552, DOI 10.17487/RFC7552, June 2015, <<https://www.rfc-editor.org/info/rfc7552>>.
- [RFC7597] Troan, O., Ed., Dec, W., Li, X., Bao, C., Matsushima, S., Murakami, T., and T. Taylor, Ed., "Mapping of Address and Port with Encapsulation (MAP-E)", RFC 7597, DOI 10.17487/RFC7597, July 2015, <<https://www.rfc-editor.org/info/rfc7597>>.
- [RFC7599] Li, X., Bao, C., Dec, W., Ed., Troan, O., Matsushima, S., and T. Murakami, "Mapping of Address and Port using Translation (MAP-T)", RFC 7599, DOI 10.17487/RFC7599, July 2015, <<https://www.rfc-editor.org/info/rfc7599>>.
- [RFC7610] Gont, F., Liu, W., and G. Van de Velde, "DHCPv6-Shield: Protecting against Rogue DHCPv6 Servers", BCP 199, RFC 7610, DOI 10.17487/RFC7610, August 2015, <<https://www.rfc-editor.org/info/rfc7610>>.

- [RFC7707] Gont, F. and T. Chown, "Network Reconnaissance in IPv6 Networks", RFC 7707, DOI 10.17487/RFC7707, March 2016, <<https://www.rfc-editor.org/info/rfc7707>>.
- [RFC7721] Cooper, A., Gont, F., and D. Thaler, "Security and Privacy Considerations for IPv6 Address Generation Mechanisms", RFC 7721, DOI 10.17487/RFC7721, March 2016, <<https://www.rfc-editor.org/info/rfc7721>>.
- [RFC7772] Yourtchenko, A. and L. Colitti, "Reducing Energy Consumption of Router Advertisements", BCP 202, RFC 7772, DOI 10.17487/RFC7772, February 2016, <<https://www.rfc-editor.org/info/rfc7772>>.
- [RFC7785] Vinapamula, S. and M. Boucadair, "Recommendations for Prefix Binding in the Context of Software Dual-Stack Lite", RFC 7785, DOI 10.17487/RFC7785, February 2016, <<https://www.rfc-editor.org/info/rfc7785>>.
- [RFC7824] Krishnan, S., Mrugalski, T., and S. Jiang, "Privacy Considerations for DHCPv6", RFC 7824, DOI 10.17487/RFC7824, May 2016, <<https://www.rfc-editor.org/info/rfc7824>>.
- [RFC7844] Huitema, C., Mrugalski, T., and S. Krishnan, "Anonymity Profiles for DHCP Clients", RFC 7844, DOI 10.17487/RFC7844, May 2016, <<https://www.rfc-editor.org/info/rfc7844>>.
- [RFC7857] Penno, R., Perreault, S., Boucadair, M., Ed., Sivakumar, S., and K. Naito, "Updates to Network Address Translation (NAT) Behavioral Requirements", BCP 127, RFC 7857, DOI 10.17487/RFC7857, April 2016, <<https://www.rfc-editor.org/info/rfc7857>>.
- [RFC7872] Gont, F., Linkova, J., Chown, T., and W. Liu, "Observations on the Dropping of Packets with IPv6 Extension Headers in the Real World", RFC 7872, DOI 10.17487/RFC7872, June 2016, <<https://www.rfc-editor.org/info/rfc7872>>.
- [RFC7915] Bao, C., Li, X., Baker, F., Anderson, T., and F. Gont, "IP/ICMP Translation Algorithm", RFC 7915, DOI 10.17487/RFC7915, June 2016, <<https://www.rfc-editor.org/info/rfc7915>>.

- [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>>.
- [RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <<https://www.rfc-editor.org/info/rfc8040>>.
- [RFC8064] Gont, F., Cooper, A., Thaler, D., and W. Liu, "Recommendation on Stable IPv6 Interface Identifiers", RFC 8064, DOI 10.17487/RFC8064, February 2017, <<https://www.rfc-editor.org/info/rfc8064>>.
- [RFC8177] Lindem, A., Ed., Qu, Y., Yeung, D., Chen, I., and J. Zhang, "YANG Data Model for Key Chains", RFC 8177, DOI 10.17487/RFC8177, June 2017, <<https://www.rfc-editor.org/info/rfc8177>>.
- [RFC8190] Bonica, R., Cotton, M., Haberman, B., and L. Vegoda, "Updates to the Special-Purpose IP Address Registries", BCP 153, RFC 8190, DOI 10.17487/RFC8190, June 2017, <<https://www.rfc-editor.org/info/rfc8190>>.
- [RFC8210] Bush, R. and R. Austein, "The Resource Public Key Infrastructure (RPKI) to Router Protocol, Version 1", RFC 8210, DOI 10.17487/RFC8210, September 2017, <<https://www.rfc-editor.org/info/rfc8210>>.
- [RFC8273] Brzozowski, J. and G. Van de Velde, "Unique IPv6 Prefix per Host", RFC 8273, DOI 10.17487/RFC8273, December 2017, <<https://www.rfc-editor.org/info/rfc8273>>.
- [RFC8343] Bjorklund, M., "A YANG Data Model for Interface Management", RFC 8343, DOI 10.17487/RFC8343, March 2018, <<https://www.rfc-editor.org/info/rfc8343>>.
- [RFC8344] Bjorklund, M., "A YANG Data Model for IP Management", RFC 8344, DOI 10.17487/RFC8344, March 2018, <<https://www.rfc-editor.org/info/rfc8344>>.
- [RFC8415] Mrugalski, T., Siodelski, M., Volz, B., Yourtchenko, A., Richardson, M., Jiang, S., Lemon, T., and T. Winters, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 8415, DOI 10.17487/RFC8415, November 2018, <<https://www.rfc-editor.org/info/rfc8415>>.

- [RFC8504] Chown, T., Loughney, J., and T. Winters, "IPv6 Node Requirements", BCP 220, RFC 8504, DOI 10.17487/RFC8504, January 2019, <<https://www.rfc-editor.org/info/rfc8504>>.
- [RFC8520] Lear, E., Droms, R., and D. Romascanu, "Manufacturer Usage Description Specification", RFC 8520, DOI 10.17487/RFC8520, March 2019, <<https://www.rfc-editor.org/info/rfc8520>>.
- [RFC8541] Litkowski, S., Decraene, B., and M. Horneffer, "Impact of Shortest Path First (SPF) Trigger and Delay Strategies on IGP Micro-loops", RFC 8541, DOI 10.17487/RFC8541, March 2019, <<https://www.rfc-editor.org/info/rfc8541>>.
- [RFC8981] Gont, F., Krishnan, S., Narten, T., and R. Draves, "Temporary Address Extensions for Stateless Address Autoconfiguration in IPv6", RFC 8981, DOI 10.17487/RFC8981, February 2021, <<https://www.rfc-editor.org/info/rfc8981>>.
- [SCANNING] Barnes, R., Altmann, R., and D. Kerr, "Mapping the Great Void - Smarter scanning for IPv6", February 2012, <http://www.caida.org/workshops/isma/1202/slides/aims1202_rbarnes.pdf>.
- [WEBER_VPN] Weber, J., "Dynamic IPv6 Prefix - Problems and VPNs", March 2018, <<https://blog.webernetz.net/wp-content/uploads/2018/03/TR18-Johannes-Weber-Dynamic-IPv6-Prefix-Problems-and-VPNs.pdf>>.

Authors' Addresses

Eric Vyncke
Cisco
De Kleetlaan 6a
Diegem 1831
Belgium

Phone: +32 2 778 4677
Email: evyncke@cisco.com

Kiran Kumar
Square
1455 Market Street, Suite 600
San Francisco 94103
United States of America

Email: kk.chittimaneni@gmail.com

Merike Kaeo
Double Shot Security
3518 Fremont Ave N 363
Seattle 98103
United States of America

Phone: +12066696394
Email: merike@doubleshotsecurity.com

Enno Rey
ERNW
Carl-Bosch-Str. 4
Heidelberg, Baden-Wuerttemberg 69115
Germany

Phone: +49 6221 480390
Email: erey@ernw.de

Internet Engineering Task Force
Internet-Draft
Intended status: BCP
Expires: March 25, 2013

J. Durand
CISCO Systems, Inc.
I. Pepelnjak
NIL
G. Doering
SpaceNet
September 21, 2012

BGP operations and security
draft-jdurand-bgp-security-02.txt

Abstract

BGP (Border Gateway Protocol) is the protocol almost exclusively used in the Internet to exchange routing information between network domains. Due to this central nature, it's important to understand the security measures that can and should be deployed to prevent accidental or intentional routing disturbances.

This document describes measures to protect the BGP sessions itself (like TTL, MD5, control plane filtering) and to better control the flow of routing information, using prefix filtering and automatization of prefix filters, max-prefix filtering, AS path filtering, route flap dampening and BGP community scrubbing.

Foreword

A placeholder to list general observations about this document.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [1].

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any

time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on March 25, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	4
2. Definitions	4
3. Protection of BGP router	4
4. Protection of BGP sessions	4
4.1. Protection of TCP sessions used by BGP	4
4.2. BGP TTL security	5
5. Prefix filtering	5
5.1. Definition of prefix filters	5
5.1.1. Prefixes that MUST not be routed by definition	5
5.1.2. Prefixes not allocated	6
5.1.3. Prefixes too specific	9
5.1.4. Filtering prefixes belonging to local AS	9
5.1.5. Internet exchange point (IXP) LAN prefixes	10
5.1.6. Default route	11
5.2. Prefix filtering recommendations in full routing networks	11
5.2.1. Filters with internet peers	12
5.2.2. Filters with customers	13
5.2.3. Filters with upstream providers	14
5.3. Prefix filtering recommendations for leaf networks	14
5.3.1. Inbound filtering	14
5.3.2. Outbound filtering	15
6. BGP route flap dampening	15
7. Maximum prefixes on a peering	15
8. AS-path filtering	16
9. Next-Hop Filtering	17
10. BGP community scrubbing	17
11. Change logs	18
11.1. Diffs between draft-jdurand-bgp-security-01 and draft-jdurand-bgp-security-00	18
11.2. Diffs between draft-jdurand-bgp-security-02 and draft-jdurand-bgp-security-01	19
12. Acknowledgements	19
13. IANA Considerations	20
14. Security Considerations	20
15. References	20
15.1. Normative References	20
15.2. Informative References	21
Authors' Addresses	22

1. Introduction

BGP [7] is the protocol used in the internet to exchange routing information between network domains. This protocol does not directly include mechanisms that control that routes exchanged conform to the various rules defined by the Internet community. This document intends to summarize most common existing rules and help network administrators applying simply coherent BGP policies.

2. Definitions

- o BGP peering: any TCP BGP connection on the Internet.

3. Protection of BGP router

The BGP router needs to be protected from stray packets. This protection should be achieved by an access-list (ACL) which would discard all packets directed to TCP port 179 on the local device and sourced from an address not known to be a BGP neighbor. If supported, an ACL specific to the control-plane of the router should be used (receive-ACL, control-plane policing, etc.), to avoid filtering transit traffic if not needed. If the hardware can not do that, interface ACLs can be used to block packets to the local router.

Some routers automatically program such an ACL upon BGP configuration. On other devices this ACL should be configured and maintained manually or using scripts.

The filtering of packets destined to the local router is a wider topic than "just for BGP" (if you bring down a router by overloading one of the other protocols from remote, BGP is harmed as well). For a more detailed recommendation, see RFC6192 [19].

4. Protection of BGP sessions

4.1. Protection of TCP sessions used by BGP

Attacks on TCP sessions used by BGP (ex: sending spoofed TCP RST packets) could bring down the TCP session. Following a successful ARP spoofing attack (or other similar Man-in-the-Middle attack), the attacker might even be able to inject packets into the TCP stream (routing attacks).

TCP sessions used by BGP can be secured with a variety of mechanisms.

MD5 protection of TCP session header [2] is the most common one, but one could also use IPsec or TCP Authentication Option (TCP-AO, [10]).

The drawback of TCP session protection is additional configuration and management overhead for authentication information (ex: MD5 password) maintenance. Protection of TCP sessions used by BGP is thus recommended when peerings are established over shared networks where spoofing can be done (like internet exchanges, IXPs).

You should block spoofed packets (packets with source IP address belonging to your IP address space) at all edges of your network, making the protection of TCP sessions used by BGP unnecessary on iBGP session or EBGP sessions run over point-to-point links.

4.2. BGP TTL security

BGP sessions can be made harder to spoof with the TTL security [9]. Instead of sending TCP packets with TTL value = 1, the routers send the TCP packets with TTL value = 255 and the receiver checks that the TTL value equals 255. Since it's impossible to send an IP packet with TTL = 255 to a non-directly-connected IP host, BGP TTL security effectively prevents all spoofing attacks coming from third parties not directly connected to the same subnet as the BGP-speaking routers.

Note: Like MD5 protection, TTL security has to be configured on both ends of a BGP session.

5. Prefix filtering

The main aspect of securing BGP resides in controlling the prefixes that are received/advertised on the BGP peerings. Prefixes exchanged between BGP peers are controlled with inbound and outbound filters that can match on IP prefixes (prefix filters, Section 5), AS paths (as-path filters, Section 8) or any other attributes of a BGP prefix (for example, BGP communities, Section 10).

5.1. Definition of prefix filters

This section list the most commonly used prefix filters. Following sections will clarify where these filters should be applied.

5.1.1. Prefixes that MUST not be routed by definition

5.1.1.1. IPv4

At the time of the writing of this document, there is no dynamic IPv4 registry listing special prefixes and their status on the internet. On the other hand static document RFC5735 [17] clarifies "special" IPv4 prefixes and their status in the Internet. Since publication of that RFC another prefix has been added on the list of the special use prefixes. Following prefixes MUST NOT cross network boundaries (ie. ASN) and therefore MUST be filtered:

- o Prefixes defined in RFC5735 [17] and more specifics
- o Shared address space [31] - 100.64.0.0/10 and more specifics

5.1.1.2. IPv6

IPv6 registry [26] maintains the list of IPv6 special purpose prefixes. With the exception of the 6to4 2002::/16 prefix in that registry, all other prefixes that are mentioned and more specifics MUST not cross network boundaries and therefore MUST be filtered. The 6to4 prefix 2002::/16 is an exception because the prefix itself can be advertised, but more specifics MUST be filtered according to [4], section 5.2.3.

At the time of the writing of this document, the list of IPv6 prefixes that MUST not cross network boundaries can be simplified as IANA allocates at the time being prefixes to RIR's only in 2000::/3 prefix [25]. All other prefixes (ULA's, link-local, multicast... are outside of that prefix) and therefore the simplified list becomes:

- o 2001:DB8::/32 and more specifics - documentation [13]
- o Prefixes more specifics than 2002::/16 - 6to4 [4]
- o 3FFE::/16 and more specifics - was initially used for the 6Bone (worldwide IPv6 test network) and returned to IANA
- o All prefixes that are outside 2000::/3 prefix

5.1.2. Prefixes not allocated

IANA allocates prefixes to RIRs which in turn allocate prefixes to LIRs. It is wise not to accept in the routing table prefixes that are not allocated. This could mean allocation made by IANA and/or allocations done by RIRs. This section details the options for building list of allocated prefixes at every level. It is important to understand that filtering prefixes not allocated requires constant updates as IANA and RIRs keep allocating prefixes. Therefore

automation of such prefix filters is key for the success of this approach. One should probably not consider solutions described in this section if it is not capable of maintaining updated prefix filters: damage would probably be worse than the intended security policy.

5.1.2.1. IANA allocated prefixes filters

IANA has allocated all the IPv4 available space. Therefore there is no reason why one would keep checking prefixes are in the IANA allocated address space [24]. No specific filter need to be put in place by administrators who want to make sure that IPv4 prefixes they receive have been allocated by IANA.

For IPv6, given the size of the address space, it can be seen as wise accepting only prefixes derived from those allocated by IANA. Administrators can dynamically build this list from the IANA allocated IPv6 space [27]. As IANA keeps allocating prefixes to RIRs, the aforementioned list should be checked regularly against changes and if they occur, prefix filter should be computed and pushed on network devices. As there is delay between the time a RIR receives a new prefix and the moment it starts allocating portions of it to its LIRs, there is no need doing this step quickly and frequently. At least process in place should make sure there is no more than one month between the time the IANA IPv6 allocated prefix list changes and the moment all IPv6 prefix filters have been updated.

If process in place (manual or automatic) cannot guarantee that the list is updated regularly then it's better not to configure any filter based on allocated networks. The IPv4 experience has shown that many network operators implemented filters for prefixes not allocated by IANA but did not update them on a regular basis. This created problems for latest allocations and required a extra work for RIR's that had to "de-boggonize" the newly allocated prefixes.

5.1.2.2. RIR allocated prefixes filters

A more precise check can be performed as one would like to make sure that prefixes they receive are being originated by the autonomous system which actually own the prefix. It has been observed in the past that one could easily advertise someone else's prefix (or more specific prefixes) and create black holes or security threats. To overcome that risk, administrators would need to make sure BGP advertisements correspond to information located in the existing registries. At this stage 2 options can be considered (short and long term options). They are described in the following subsections.

5.1.2.3. Prefix filters creation from Internet Routing Registries (IRR)

An Internet Routing Registry (IRR) is a database containing internet routing information, described using Routing Policy Specification Language objects [14]. Network engineers are given privileges to describe routing policies of their own networks in the IRR and information is published, usually publicly. Most of Regional Internet Registries do also operate an IRR and can control that registered routes conform to allocations made.

It is possible to use IRR information in order to build for a given BGP neighbor a list of prefixes, with corresponding originating autonomous system. This can be done relatively easily using scripts and existing tools capable of retrieving this information in the registries. This approach is exactly the same for both IPv4 and IPv6.

The macro-algorithm for the script is described as follows. For the peer that is considered, the distant network administrator has provided the autonomous system and may be able to provide an AS-SET object (aka AS-MACRO). An AS-SET is an object which contains AS numbers or other AS-SET's. An operator may create an AS-SET defining all the AS numbers of its customers. A tier 1 transit provider might create an AS-SET describing the AS-SET of connected operators, which in turn describe the AS numbers of their customers. Using recursion, it is possible to retrieve from an AS-SET the complete list of AS numbers that the peer is susceptible to announce. For each of these AS numbers, it is also easy to check in the corresponding IRR all associated prefixes. With these 2 mechanisms a script can build for a given peer the list of allowed prefixes and the AS number from which they should be originated.

As prefixes, AS numbers and AS-SET's may not all be under the same RIR authority, a difficulty resides choosing for each object the appropriate IRR to poll. Some IRR have been created and are not restricted to a given region or authoritative RIR. They allow RIRs to publish information contained in their IRR in a common place. They also make it possible for any subscriber (probably under contract) to publish information too. When doing requests inside such an IRR, it is possible to specify the source of information in order to have the most reliable data. One could check the central registry and only check that the source is one of the 5 RIRs. The probably most famous registry of that kind is the RADB [28] (Routing Assets Database).

As objects in IRR's may quickly vary over time, it is important that prefix filters computed using this mechanism are refreshed regularly. A daily basis could even be considered as some routing changes must

be done sometimes in a certain emergency and registries may be updated at the very last moment. It has to be noted that this approach significantly increases the complexity of the router configurations as it can quickly add more than ten thousands configuration lines for some important peers.

5.1.2.4. SIDR - Secure Inter Domain Routing

IETF has created a working group called SIDR (Secure Inter-Domain Routing) in order to create an architecture to secure internet advertisements. At the time this document is written, many document has been published and a framework is proposed so that advertisements can be checked against signed routing objects in RIR routing registries. Implementing mechanisms proposed by this working group is the solution that will solve at a longer term the BGP routing security. But as it may take time objects are signed and deployments are done such a solution will need to be combined at the time being with other mechanisms proposed in this document. The rest of this section assumes the reader understands all technologies associated with SIDR.

Each received route on a router should be checked against the RPKI data set: if a corresponding ROA is found and is valid then the prefix should be accepted. If the ROA is found and is INVALID then the prefix should be discarded. If an ROA is not found then the prefix should be accepted but corresponding route should be given a low preference.

5.1.3. Prefixes too specific

Most ISPs will not accept advertisements beyond a certain level of specificity (and in return do not announce prefixes they consider as too specific). That acceptable specificity is decided for each peering between the 2 BGP peers. Some ISP communities have tried to document acceptable specificity. This document does not make any judgement on what the best approach is, it just recalls that there are existing practices on the internet and recommends the reader to refer to what those are. As an example RIPE community has documented that IPv4 prefixes longer than /24 and IPv6 prefixes longer than /48 are generally not announced/accepted in the internet [21] [22].

5.1.4. Filtering prefixes belonging to local AS

A network SHOULD filter its own prefixes on peerings with all its peers (inbound direction). This prevents local traffic (from a local source to a local destination) to leak over an external peering in case someone else is announcing the prefix over the Internet. This also protects the infrastructure which may directly suffer in case

backbone's prefix is suddenly preferred over the Internet. To an extent, such filters can also be configured on a network for the prefixes of its downstreams in order to protect them too. Such filters must be defined with caution as they can break existing redundancy mechanisms. For example in case an operator has a multihomed customer, it should keep accepting the customer prefix from its peers and upstreams. This will make it possible for the customer to keep accessing its operator network (and other customers) via the internet in case the BGP peering between the customer and the operator is down.

5.1.5. Internet exchange point (IXP) LAN prefixes

5.1.5.1. Network security

When a network is present on an exchange point (IXP) and peers with other IXP members over a common subnet (IXP LAN prefix), it **MUST NOT** accept more specific prefixes for the IXP LAN prefix from any of all its external BGP peers. Accepting these routes would create a black hole for connectivity to the IXP LAN.

If the IXP LAN prefix is accepted as an "exact match", care needs to be taken to avoid other routers in the network sending IXP traffic towards the externally-learned IXP LAN prefix (recursive route lookup pointing into the wrong direction). This can be achieved by preferring IGP routes before eBGP, or by using "BGP next-hop-self" on all routes learned on that IXP.

If the IXP LAN prefix is accepted at all, it **MUST** only be accepted from the ASes that the IXP authorizes to announce it - which will usually be automatically achieved by filtering announcements by IRR DB.

5.1.5.2. pMTUd and loose uRPF problem

In order to have pMTUd working in the presence of loose uRPF, it is necessary that all the networks that may source traffic that could flow through the IXP (ie. IXP members and their downstreams) have a route for the IXP LAN prefix. This is necessary as "packet too big" ICMP messages sent by IXP members' routers may be sourced using an address of the IXP LAN prefix. In the presence of loose uRPF, this ICMP packet is dropped if there is no route for the IXP LAN prefix or a less specific route covering IXP LAN prefix.

In that case, any IXP member **SHOULD** make sure it has a route for the IXP LAN prefix or a less specific prefix on all its routers and that it announces the IXP LAN prefix or less specific (up to a default route) to its downstreams. The announcements done for this purpose

SHOULD pass IRR-generated filters described in Section 5.1.2.3 as well as "prefixes too specific" filters described in Section 5.1.3. The easiest way to implement this is that the IXP itself takes care of the origination of its prefix and advertises it to all IXP members through a BGP peering. Most likely the BGP route servers would be used for this. The IXP would most likely send its entire prefix which would be equal or less specific than the IXP LAN prefix.

5.1.5.3. Example

Let's take as an example an IXP in RIPE region for IPv4. It would be allocated a /22 by RIPE NCC (X.Y.0.0/22 in our example) and use a /23 of this /22 for the IXP LAN (let say X.Y.0.0/23). This IXP LAN prefix is the one used by IXP members to configure eBGP peerings. The IXP could also be allocated an AS number (AS64496 in our example).

Any IXP member MUST make sure it filters prefixes more specific than X.Y.0.0/23 from all its eBGP peers. If it received X.Y.0.0/24 or X.Y.1.0/24 this could seriously impact its routing.

The IXP SHOULD originate X.Y.0.0/22 and advertise it to its members through its BGP route servers (configured with AS64496).

The IXP members SHOULD accept the IXP prefix only if it passes the IRR generated filters (see Section 5.1.2.3)

IXP members SHOULD then advertise X.Y.0.0/22 prefix to their downstreams. This announce would pass IRR based filters as it is originated by the IXP.

5.1.6. Default route

5.1.6.1. IPv4

0.0.0.0/0 prefix MUST NOT be announced on the Internet but it is usually exchanged on upstream/customer peerings.

5.1.6.2. IPv6

::/0 prefix MUST NOT be announced on the Internet but it is usually exchanged on upstream/customer peerings.

5.2. Prefix filtering recommendations in full routing networks

For networks that have the full internet BGP table, some policies should be applied on each BGP peer for received and advertised routes. It is recommended that each autonomous system configures

rules for advertised and received routes at all its borders as this will protect the network and its peer even in case of misconfiguration. The most commonly used filtering policy is proposed in this section.

5.2.1. Filters with internet peers

5.2.1.1. Inbound filtering

There are basically 2 options, the loose one where no check will be done against RIR allocations and the strict one where it will be verified that announcements strictly conform to what is declared in routing registries.

5.2.1.1.1. Inbound filtering loose option

In that case, the following prefixes received from a BGP peer will be filtered:

- o Prefixes not routable (Section 5.1.1)
- o Prefixes not allocated by IANA (IPv6 only) (Section 5.1.2.1)
- o Routes too specific (Section 5.1.3)
- o Prefixes belonging to local AS (Section 5.1.4)
- o Exchange points LAN prefixes (Section 5.1.5)
- o Default route (Section 5.1.6)

5.2.1.1.2. Inbound filtering strict option

In that case, filters are applied to make sure advertisements strictly conform to what is declared in routing registries Section 5.1.2.2. It must be checked that in case of script failure all routes are rejected.

In addition to this, one could apply following filters beforehand in case routing registry used as source of information by the script is not fully trusted:

- o Prefixes not routable (Section 5.1.1)
- o Routes too specific (Section 5.1.3)
- o Prefixes belonging to local AS (Section 5.1.4)

- o Exchange points LAN prefixes (Section 5.1.5)
- o Default route (Section 5.1.6)

5.2.1.2. Outbound filtering

Configuration in place will make sure that only appropriate prefixes are sent. These can be for example prefixes belonging to the considered networks and those of its customers. This can be done using BGP communities or many other solution. Whatever scenario considered, it can be desirable that following filters are positioned before to avoid unwanted route announcement due to bad configuration:

- o Prefixes not routable (Section 5.1.1)
- o Routes too specific (Section 5.1.3)
- o Exchange points LAN prefixes (Section 5.1.5)
- o Default route (Section 5.1.6)

In case it is possible to list the prefixes to be advertised, then just configuring the list of allowed prefixes and denying the rest is sufficient.

5.2.2. Filters with customers

5.2.2.1. Inbound filtering

Inbound policy with end customers is pretty straightforward: only customers prefixes must be accepted, all others MUST be discarded. The list of accepted prefixes can be manually specified, after having verified that they are valid. This validation can be done with the appropriate IP address management authorities.

Same rules apply in case the customer is also a network connecting other customers (for example a tier 1 transit provider connecting service providers). An exception can be envisaged in case it is known that the customer network applies strict inbound/outbound prefix filtering, and the number of prefixes announced by that network is too large to list them in the router configuration. In that case filters as in Section 5.2.1.1 can be applied.

5.2.2.2. Outbound filtering

Outbound policy with customers may vary according to the routes customer wants to receive. In the simplest possible scenario, customer wants to receive only the default route, which can be done

easily by applying a filter with the default route only.

In case the customer wants to receive the full routing (in case it is multihomed or if wants to have a view on the internet table), the following filters can be simply applied on the BGP peering:

- o Prefixes not routable (Section 5.1.1)
- o Routes too specific (Section 5.1.3)
- o Default route (Section 5.1.6)

There can be a difference for the default route that can be announced to the customer in addition to the full BGP table. This can be done simply by removing the filter for the default route. As the default route may not be present in the routing table, one may decide to originate it only for peerings where it has to be advertised.

5.2.3. Filters with upstream providers

5.2.3.1. Inbound filtering

In case the full routing table is desired from the upstream, the prefix filtering to apply is more or less the same than the one for peers Section 5.2.1.1. There can be a difference for the default route that can be desired from an upstream provider even if it advertises the full BGP table. In case the upstream provider is supposed to announce only the default route, a simple filter will be applied to accept only the default prefix and nothing else.

5.2.3.2. Outbound filtering

The filters to be applied should not differ from the ones applied for internet peers (Section 5.2.1.2).

5.3. Prefix filtering recommendations for leaf networks

5.3.1. Inbound filtering

The leaf network will position the filters corresponding to the routes it is requesting from its upstream. In case a default route is requested, simple inbound filter will be applied to accept only that default route (Section 5.1.6). In case the leaf network is not capable of listing the prefix because the amount is too large (for example if it requires the full internet routing table) then it should configure filters to avoid receiving bad announcements from its upstream:

- o Prefixes not routable (Section 5.1.1)
- o Routes too specific (Section 5.1.3)
- o Prefixes belonging to local AS (Section 5.1.4)
- o Default route (Section 5.1.6) depending if the route is requested or not

5.3.2. Outbound filtering

A leaf network will most likely have a very straightforward policy: it will only announce its local routes. It can also configure the following prefixes filters described in Section 5.2.1.2 to avoid announcing invalid routes to its upstream provider.

6. BGP route flap dampening

BGP route flap dampening mechanism makes it possible to give penalties to routes each time they change in the BGP routing table. Initially this mechanism was created to protect the entire internet from multiple events impacting a single network. RIPE community now recommends not using BGP route flap dampening [20]. Author of this document proposes to follow the proposal of the RIPE community.

7. Maximum prefixes on a peering

It is recommended to configure a limit on the number of routes to be accepted from a peer. Following rules are generally recommended:

- o From peers, it is recommended to have a limit lower than the number of routes in the internet. This will shut down the BGP peering if the peer suddenly advertises the full table. One can also configure different limits for each peer, according to the number of routes they are supposed to advertise plus some headroom to permit growth.
- o From upstreams which provide full routing, it is recommended to have a limit much higher than the number of routes in the internet. A limit is still useful in order to protect the network (and in particular the routers' memory) if too many routes are sent by the upstream. The limit should be chosen according to the number of routes that can actually be handled by routers.

It is important to regularly review the limits that are configured as the internet can quickly change over time. Some vendors propose

mechanisms to have 2 thresholds: while the higher number specified will shutdown the peering, the first threshold will only trigger a log and can be used to passively adjust limits based on observations made on the network.

8. AS-path filtering

The following rules should be applied on BGP AS-paths:

- o Do not accept anything other than customer's AS number from the customer. Alternatively, only accept AS-paths with a single AS number (potentially repeated several times) from your customers. The latter option is easier to configure than per-customer AS-path filters: the default BGP logic will make sure in that case that the first AS number in the AS-path is the one of the peer.
- o Do not accept overly long AS path prepending from the customer.
- o Do not accept more than two distinct AS path numbers in the AS path if your customer is an ISP with customers. This rule is not adding anything extra in case prefix filters are built from registries as described in Section 5.1.2.3.
- o Do not advertise prefixes with non-empty AS-path if you're not transit.
- o Do not advertise prefixes with upstream AS numbers in the AS path to your peering AS.
- o Do not accept private AS numbers except from customers
- o Do not advertise private AS numbers. Exception: Customers using BGP without having their own AS number must use private AS numbers to advertise their prefixes to their upstream. The private AS number is usually provided by the upstream.
- o Do not accept prefixes when the first AS number in the AS-path is not the one of the peer. In case the peering is done toward a BGP route-server [30] (connection on an Internet eXchange Point - IXP) with transparent AS path handling, this verification needs to be de-activated as the first AS number will be the one of an IXP member whereas the peer AS number will be the one of the BGP route-server.

9. Next-Hop Filtering

If peering on a shared network, like an Exchange-Point, BGP can advertise prefixes with a 3rd-party next-hop, thus directing packets not to the peer announcing the prefix but somewhere else.

This is a desirable property for BGP route-server setups [30], where the route-server will relay routing information, but has neither capacity nor desire to receive the actual data packets. So the BGP route-server will announce prefixes with a next-hop setting pointing to the router that originally announced the prefix to the route-server.

In direct peerings between ISPs, this is undesirable, as one of the peers could trick the other one to send packets into a black hole (unreachable next-hop) or to an unsuspecting 3rd party who would then have to carry the traffic. Especially for black-holing, the root cause of the problem is hard to see without inspecting BGP prefixes at the receiving router at the IXP.

Therefore, the authors recommend to, by default, apply an inbound route policy to IXP peerings which sets the next-hop for accepted prefixes to the BGP peer that sent the prefix (which is what "next-hop-self" would enforce on the sending side, but you can not rely on the other party to always send correct information).

This policy **MUST NOT** be used on route-server peerings, or on peerings where you intentionally permit the other side to send 3rd-party next-hops.

10. BGP community scrubbing

Optionally we can consider the following rules on BGP AS-paths:

- o Scrub inbound communities with your AS number in the high-order bits - allow only those communities that customers/peers can use as a signaling mechanism
- o Do not remove other communities: your customers might need them to communicate with upstream providers. In particular do not (generally) remove the no-export community as it is usually announced by your peer for a certain purpose.

11. Change logs

11.1. Diffs between draft-jdurand-bgp-security-01 and draft-jdurand-bgp-security-00

Following changes have been made since previous document draft-jdurand-bgp-security-00:

- o "This documents" typo corrected in the former abstract
- o Add normative reference for RFC5082 in former section 3.2
- o "Non routable" changed in title of former section 4.1.1
- o Correction of typo for IPv4 loopback prefix in former section 4.1.1.1
- o Added shared transition space 100.64.0.0/10 in former section 4.1.1.1
- o Clarification that 2002::/16 6to4 prefix can cross network boundaries in former section 4.1.1.2
- o Rationale of 2000::/3 explained in former section 4.1.1.2
- o Added 3FFE::/16 prefix forgotten initially in the simplified list of prefixes that MUST not be routed by definition in former section 4.1.1.2
- o Warn that filters for prefixes not allocated by IANA must only be done if regular refresh is guaranteed, with some words about the IPv4 experience, in former section 4.1.2.1
- o Replace RIR database with IRR. A definition of IRR is added in former section 4.1.2.2
- o Remove any reference to anti-spoofing in former section 4.1.4
- o Clarification for IXP LAN prefix and pMTUd problem in former section 4.1.5
- o "Autonomous filters" typo (instead of Autonomous systems) corrected in the former section 4.2
- o Removal of an example for manual address validation in former section 4.2.2.1

- o RFC5735 obsoletes RFC3300
 - o Ingress/Egress replaced by Inbound/Outbound in all the document
- 11.2. Diffs between draft-jdurand-bgp-security-02 and draft-jdurand-bgp-security-01

Following changes have been made since previous document draft-jdurand-bgp-security-01:

- o 2 documentation prefixes were forgotten due to errata in RFC5735. But all prefixes were removed from that document which now point to other references for sake of not creating a new "registry" that would become outdated sooner or later.
- o Change MD5 section with global TCP security session and introducing TCP-AO in former section 3.1. Added reference to BCP38
- o Added new section 3 about BGP router protection with forwarding plane ACL
- o Change text about prefix acceptable specificity in former section 4.1.3 to explain this doc does not try to make recommendations
- o Refer as much as possible to existing registries to avoid creating a new one in former section 4.1.1.1 and 4.1.1.2
- o Abstract reworded
- o 6to4 exception described (only more specifics must be filtered)
- o More specific -> more specifics
- o should -> MUST for the prefixes an ISP needs to filter from its customers in former section 4.2.2.1
- o Added "plus some headroom to permit growth" in former section 7
- o Added new section on Next-Hop filtering

12. Acknowledgements

Authors would like to thank the following people for their comments and support: Marc Blanchet, Ron Bonica, Daniel Ginsburg, David Groves, Tim Kleefass, Hagen Paul Pfeifer, Thomas Pinaud, Carlos Pignataro, Matjaz Straus, Tony Tauber, Gunter Van de Velde, Sebastian

Wiesinger.

13. IANA Considerations

This memo includes no request to IANA.

14. Security Considerations

This document is entirely about BGP operational security.

15. References

15.1. Normative References

- [1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <<http://xml.resource.org/public/rfc/html/rfc2119.html>>.
- [2] Heffernan, A., "Protection of BGP Sessions via the TCP MD5 Signature Option", RFC 2385, August 1998.
- [3] Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629, June 1999.
- [4] Carpenter, B. and K. Moore, "Connection of IPv6 Domains via IPv4 Clouds", RFC 3056, February 2001.
- [5] Huitema, C. and B. Carpenter, "Deprecating Site Local Addresses", RFC 3879, September 2004.
- [6] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast Addresses", RFC 4193, October 2005.
- [7] Rekhter, Y., Li, T., and S. Hares, "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, January 2006.
- [8] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, February 2006.
- [9] Gill, V., Heasley, J., Meyer, D., Savola, P., and C. Pignataro, "The Generalized TTL Security Mechanism (GTSM)", RFC 5082, October 2007.
- [10] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", RFC 5925, June 2010.

15.2. Informative References

- [11] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", RFC 2234, November 1997.
- [12] 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.
- [13] Huston, G., Lord, A., and P. Smith, "IPv6 Address Prefix Reserved for Documentation", RFC 3849, July 2004.
- [14] Blunk, L., Damas, J., Parent, F., and A. Robachevsky, "Routing Policy Specification Language next generation (RPSLng)", RFC 4012, March 2005.
- [15] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", RFC 4234, October 2005.
- [16] Blanchet, M., "Special-Use IPv6 Addresses", RFC 5156, April 2008.
- [17] Cotton, M. and L. Vegoda, "Special Use IPv4 Addresses", BCP 153, RFC 5735, January 2010.
- [18] Arkko, J., Cotton, M., and L. Vegoda, "IPv4 Address Blocks Reserved for Documentation", RFC 5737, January 2010.
- [19] Dugal, D., Pignataro, C., and R. Dunn, "Protecting the Router Control Plane", RFC 6192, March 2011.
- [20] Smith, P. and C. Panig1, "RIPE-378 - RIPE Routing Working Group Recommendations On Route-flap Damping", May 2006.
- [21] Smith, P., Evans, R., and M. Hughes, "RIPE-399 - RIPE Routing Working Group Recommendations on Route Aggregation", December 2006.
- [22] Smith, P. and R. Evans, "RIPE-532 - RIPE Routing Working Group Recommendations on IPv6 Route Aggregation", November 2011.
- [23] Doering, G., "IPv6 BGP Filter Recommendations", November 2009, <<http://www.space.net/~gert/RIPE/ipv6-filters.html>>.
- [24] "IANA IPv4 Address Space Registry", <<http://www.iana.org/assignments/ipv4-address-space/ipv4-address-space.xml>>.
- [25] "IANA IPv6 Address Space", <<http://www.iana.org/assignments/>>

ipv6-address-space/ipv6-address-space.xml>.

- [26] "IANA IPv6 Special Purpose Registry", <<http://www.iana.org/assignments/iana-ipv6-special-registry/iana-ipv6-special-registry.xml>>.
- [27] "IANA IPv6 Address Space Registry", <<http://www.iana.org/assignments/ipv6-unicast-address-assignments/ipv6-unicast-address-assignments.xml>>.
- [28] "Routing Assets Database", <<http://www.radb.net>>.
- [29] "Secure Inter-Domain Routing IETF working group", <<http://datatracker.ietf.org/wg/sidr/>>.
- [30] "Internet Exchange Route Server", <<http://tools.ietf.org/id/draft-jasinska-ix-bgp-route-server-03.txt>>.
- [31] "IANA Reserved IPv4 Prefix for Shared Address Space", <<http://tools.ietf.org/id/draft-weil-shared-transition-space-request-15.txt>>.

Authors' Addresses

Jerome Durand
CISCO Systems, Inc.
11 rue Camille Desmoulins
Issy-les-Moulineaux 92782 CEDEX
FR

Email: jerduran@cisco.com

Ivan Pepelnjak
NIL Data Communications
Tivolska 48
Ljubljana 1000
Slovenia

Email: ip@nil.com

Gert Doering
SpaceNet AG
Joseph-Dollinger-Bogen 14
Muenchen D-80807
Germany

Email: gert@space.net

