March 2019 – Version 1.0

Network Security

IPv6 Security for IPv4 Engineers

Author
Fernando Gont

Internet Society

Abstract

This document provides an overview of IPv6 security that is specifically aimed at IPv4 engineers and operators. Rather than describing IPv6 in an isolated manner, it aims to re-use as much of the existing IPv4 knowledge and experience as possible. It highlights the security issues that affect both protocols in the same manner, as well as those that are new or different for the IPv6 protocol suite. Additionally, it discusses the security implications arising from the co-existence of the IPv6 and IPv4 protocols.

1. Introduction

The IPv6 protocol suite has been designed to accommodate the present and future growth of the Internet by providing a much larger address space than that of its IPv4 counterpart, and is expected to be the successor of the original IPv4 protocol suite. The imminent exhaustion of the IPv4 address space has already led to the deployment of IPv6 in a large number of production environments, with many other organizations planning to deploy IPv6 in the short or near term.

There are a number of factors that make the IPv6 protocol suite interesting from a security standpoint [IPV6-SEC]. Firstly, being a newer technology, technical personnel have less confidence with the IPv6 protocol suite than with its IPv4 counterpart, and it is therefore possible that the security implications of the protocols will be overlooked when they are deployed on production networks. Secondly, IPv6 implementations tend to be less mature than their IPv4 counterparts, and it therefore likely that vulnerabilities will be discovered before their robustness matches that of existing IPv4 implementations.Thirdly, security products such as firewalls and NIDS's (Network Intrusion Detection Systems) usually have less support for the IPv6 protocols than for their IPv4 counterparts [IPV6-SEC-2] [IPV6-FW]. Fourthly, the security implications of IPv6 transition/co-existence technologies on existing IPv4 networks are usually overlooked, potentially enabling attackers to leverage these technologies to circumvent IPv4 security controls in unexpected ways [RFC7123]. Finally, marketing claims have created a lot of myths around IPv6 (and IPv6 security in particular), hindering proper awareness about IPv6 and its security considerations in particular [IPV6-MYTHS].

This document provides an overview of IPv6 security that is specifically aimed at IPv4 engineers and operators. Rather than describing IPv6 in an isolated manner, it aims to re-use as much of the existing IPv4 knowledge and experience as possible, by highlighting the security issues that affect both protocols in the same manner, those that are new or different for the IPv6 protocol suite, and those that arise from the co-existence of IPv6 and IPv4.

2. Security Implications of the IPv6 Protocol Suite

The IPv6 protocol suite provides very similar functionality and services to those provided by its IPv4 counterpart. Namely, it provides an unreliable datagram transfer service to the upper layers and implements features such as automatic host configuration, fault isolation, etc. However, most of these features are implemented by means of completely different mechanisms.

The following table provides a rough comparison of IPv4 and IPv6 features, highlighting how each feature is implemented in each version of the protocol:

Feature	IPv4	IPv6
Addressing	32 bits	128 bits
Packet structure	Variable-length header	Fixed-size base header + Extension Headers
Address Resolution	ARP	ICMPv6 NS/NA (+ MLD)
Auto-configuration	DHCP	ICMPv6 RS/RA & DHCPv6 (+ MLD)
Fault Isolation	ICMPv4	ICMPv6
IPsec support	Optional	Optional
Fragmentation	Both in hosts and routers	Only in hosts
Multicast Usage	Only for Multicast apps	Required for Neighbor Discovery
Network Architecture	Private addresses + NAT	Global addresses + Firewall

Comparing the IPv4 and IPv6 protocol suites in this manner is particularly important from a security standpoint for at less two reasons. Firstly, it stresses what features/services are present in both protocols, and consequently what features/services can be subject to attacks. Secondly, spelling out the specific mechanisms that are employed to implement a given feature/service provides a hint regarding the possible differences in corresponding attacks, as well as in possible mitigations.

The following subsections provide an overview of each of the aforementioned IPv6 features, how the implementation of each feature differs from the IPv4 counterpart, and what are the security implications of such differences.

2.1. IP Addressing

The main driver for the adoption and deployment of IPv6 is its larger address space. The structure

of IPv6 addresses is quite similar to that of their IPv4 counterparts, namely there are different address types (unicast, multicast, etc.) and scopes (link-local, global, etc.) and addresses are normally aggregated (as "prefixes") for the purpose of routing. However, IPv6 addresses are obviously much larger than their IPv4 counterparts: 128-bits vs. 32 bits. Similarly, IPv6 subnets are normally much larger than their IPv4 counterparts: in IPv4, /24 (or smaller) subnets are quite common, while /64 subnets are the norm for the IPv6 case, allowing for 2^{64} possible addresses - a virtually unlimited number of addresses for each subnet.

While IPv4 systems typically employ only one address per network interface, IPv6 systems normally employ multiple addresses for each network interface. For example, each network interface will normally employ one link-local unicast address, and one (or more) global unicast address. IPv6 also makes extensive use of multicasting (e.g. for address resolution and automatic configuration) and nodes are therefore normally reachable via one or more multicast addresses.

These differences warrant the discussion of a number of topics that are closely related to IPv6 addressing:

- IPv6 network reconnaissance
- Impact of IPv6 subnet size on IPv6 stack resiliency
- Challenges arising from IPv6 host address availability
- Lack of Address Translation

The following sub-sections discuss each of these areas, and their corresponding security implications.

2.1.1. IPv6 Network Reconnaissance

The much larger IPv6 subnet size results in a much lower host address density in IPv6 subnets. That is, given an IPv6 subnet, only a very small fraction of the available addresses are actually employed by nodes on such subnet (see [RFC4692] and [RFC7421] for further discussion). If addresses are randomly distributed over the 2^{64} possible values, it becomes unfeasible to brute-force search for "alive" nodes on a target network (as with the typical "ping sweeps" normally employed in the IPv4 world). This may be beneficial for mitigating network reconnaissance attacks but may also hinder legitimate penetration tests and security assessments.

The feasibility and effectiveness of IPv6 address scans depends on whether the addresses of nodes in the target network follow specific patterns. For example, recent versions of most

operating systems generate unpredictable IPv6 addresses when SLAAC is employed (see [RFC8064] and [RFC7217]).

In such scenarios, traditional brute-force network reconnaissance via the so-called "ping sweeps" becomes unfeasible, and thus alternative techniques need to be employed. However, servers, routers, and other infrastructure systems tend to employ manual configuration and typically result in predictable addresses that may be easily discovered by means of IPv6 address scans [RFC7707].

The effectiveness of IPv6 address scans will typically depend on how addresses are configured (SLAAC, DHCPv6, or manual configuration) and the type of target node (clients, servers, routers, etc.). [RFC7707] discusses network reconnaissance in IPv6 networks in detail.

While there is a plethora of techniques for performing network reconnaissance in IPv6 networks, some of them have been found to be particularly effective.

If the target is a local subnet, the following techniques have been found to be effective:

- Multicasted probes (ICMPv6 echo, and specially-crafted probe packets that elicit ICMPv6 error messages)
- Multicast DNS (mDNS) queries.

On the other hand, if the target is a remote network, the following techniques may be used:

- Pattern-based address scans
- DNS zone transfers
- DNS reverse mappings
- Certificate transparency framework
- Search engines

Pattern-based address scans are discussed in detail in [RFC7707], whilst [DNS-REV] and [DNS-REV2] discuss DNS reverse mappings with practical examples. [CTF] discusses the use of search engines and the certificate transparency framework for network reconnaissance, with practical examples. [IPV6-REC] provides practical examples of several techniques for IPv6 network reconnaissance using open source tools.

We note that whilst the address scan techniques are IPv6-specific, the other techniques can also be employed to map IPv4 systems and networks. However, they become more relevant for IPv6 because of the unfeasibility to perform traditional brute-force address scans.

2.1.2. Impact of IPv6 subnet size on IPv6 stack resiliency

IPv4 subnets do not accommodate more than a few hundred or thousand nodes, and the small IPv4 subnet size artificially limits the size of internal protocol data structures that store information about on-link nodes (such as the ARP cache). On the other hand, the standard IPv6 subnet size allows for a virtually unlimited number of addresses in each subnet, and therefore does not enforce any artificial limits on the aforementioned data structures. This means that, unless implementations enforce explicit limits on such data structures, an attacker could cause such structures to grow without bounds, possibly leading to Denial of Service (DoS) situations.

For example, the impact of the IPv6 subnet size on the Neighbor Cache (NC) has been discussed in [RFC6583] and [OPSEC-ND]. The associated issues can (and should) be mitigated by the IPv6 implementations enforcing appropriate limits on the maximum number of NC entries in the "INCOMPLETE" state. However, when such mitigations are not readily available, operational mitigations such as employing smaller subnets for point-to-point links (see [RFC6164]) could be employed.

Similar issues may affect the data structures storing configured IPv6 addresses and IPv6 routes. [ND-ATTCK] discusses some of these attacks and provides practical examples.

2.1.3. Challenges arising from IPv6 host address availability

IPv6 nodes typically configure multiple addresses for each network interface. For example, a given node may configure both stable [RFC7217] and temporary [RFC4941] addresses for each prefix advertised via SLAAC [RFC4862] for address configuration. Some of these addresses (e.g. temporary addresses) may have a limited lifetime and change over time. This may have a number of operational implications.

While network devices should be prepared to handle multiple addresses per node (in terms of Neighbor Cache entries Multicast Listener Discovery groups, etc.), this might not be the case. For example, some network devices might not be prepared to handle so many addresses or might infer that the use of multiple addresses by single node is the result of source address "spoofing".

In scenarios where temporary addresses are employed, it may be difficult to correlate network activity. If such correlation is deemed necessary, an operator may need to employ ad-hoc mechanisms (see e.g. [NDPMON]) to maintain a centralized log that records which addresses have been employed by which node at which point in time, disable temporary addresses [BROERSMA], and/or employ DHCPv6 for address configuration. This is in contrast with the IPv4 case, where addresses are typically leased by a DHCP server, and hence a centralized log of network addresses is normally maintained.

Use of multiple addresses per node is normally encouraged, and nodes are generally free to configure as many addresses as deemed necessary [RFC7934]. This means that network devices should be prepared to handle and allow multiple addresses per node. Security devices enforcing IPv6 ACLs may need to do so on a per-prefix basis rather than on a per-address basis.

For example, it would be virtually no use for a network security device to try to protect a node by limiting the number of incoming connections on a per-address basis, since nodes have virtually no limit on the addresses the may configure and use. If such policies are to be enforced, security devices may want to limit the number of incoming connections on a per-prefix basis, instead.

The ability to use multiple addresses per node may be beneficial for a number of reasons. For example, a network could use both global unicast addresses (GUAs) and unique-local unicast addresses (ULAs) [RFC4291] such that global addresses are employed when communicating with external nodes, whilst ULAs are employed when communicating with internal nodes. Thus, any communication between internal nodes could survive connectivity outages that might prevent the network from employing global addresses. Additionally, use of only ULAs for nodes that should only be reachable within the internal network may provide an additional layer of isolation (in addition to proper packet filtering where appropriate) [FIREWALLS].

If temporary addresses are employed, host and network firewalls should generally be configured such that outgoing communications are allowed from any address, but incoming communications are only allowed to stable addresses (as opposed to all available addresses). Thus, IPv6 addresses exposed when communicating with external nodes will not result in nodes being exposed to unsolicited incoming communications and attacks. For obvious reasons, nodes that do not expect incoming communications should reject incoming communications to both stable and temporary addresses (i.e., to all addresses).

Operational and security considerations arising from the use of IPv6 addresses are discussed in detail in [IPV6-ADDR].

2.1.4. Lack of Address Translation

Since the motivation for IPv6 deployment is its larger address space, it is expected that the vast majority of IPv6 networks will not employ any kind of Network Address Translation (NAT). In scenarios where networks employ Provider Assigned (PA) address space, any changes in the network prefix delegated by the ISP would result in a renumbering event for the whole network.

Additionally, in the event of network outages affecting the connectivity with the upstream network provider for an extended period of time, the network might be forced to stop using any prefixes previously delegated/leased by the upstream provider. This might result in an outage of internal network communications if such address space is the only one employed.

Unique Local Addresses [RFC4193] are, roughly, the equivalent of IPv4 private addresses [RFC1918] in the IPv6 world. ULA prefixes are not assigned or leased by the upstream ISP, but rather selected by the local network administrator, and therefore will normally not be subject of renumbering events -- hence requiring fewer updates to the DNS, ACLs, etc. Additionally, since they are not leased/delegated by an upstream ISP, even if there is a network outage affecting the connectivity with the upstream ISP for an extended period of time, ULAs may still be used for communicating with internal nodes.

ULA prefixes can be particularly beneficial for nodes that should not be reachable from external networks, since:

- the “private” nature of ULAs provides an additional layer of isolation (other than appropriate packet filtering) because these addresses are unlikely to be reachable from external networks
- they may result in addresses that are more stable than those configured for prefixes leased/delegated by an upstream provider
- they allow network operation even in the presence of outages in the upstream connectivity.

In most scenarios, ULAs will be configured alongside global unicast addresses, with ULAs being used for internal communications, while global addresses will be used for communication with external nodes.

2.2. IP Packet Structure

IPv4 employs a variable-length packet header that can grow (up to 64 bytes) to accommodate IPv4 options. Since the same "container" is employed to carry all types of options, all nodes are required (at least in theory) to parse all IPv4 options looking for options they might need to process.

IPv6 instead employs a fixed-length base IPv6 header with optional "extension headers" that form a "daisy-chain" packet structure. Different option "containers" are employed depending on which systems are expected to process the options, such that nodes are not forced to parse options they are not expected to process. However, the IPv6 packet structure tends to be unfriendly with modern router architectures when the entire IPv6 header chain needs to be processed to access upper-layer protocol values (such as transport protocol type, transport protocol port numbers, etc.) [RTR-ARCH] [IPV6-EHS].

There are a number of security implications arising from IPv6 extension headers:

- Some security devices fail to process the entire IPv6 header chain when enforcing a filtering policy (see e.g., [RFC7113]). As a result, even the simple addition of an extension header that carries only "padding" options may be enough to circumvent the corresponding security controls.

- Some network and/or security devices may normally process traffic in hardware, but resort to process packets carrying options in software. In such scenarios, IPv6 extension headers may be leveraged to perform DoS (denial of service) attacks.

- Many IPv6 implementations have been found to fail to perform basic sanity checks on packets employing IPv6 extension headers. In some cases, an attacker may cause a processing node to crash, reboot, or become unresponsive by sending either a single or a sustained flow of crafted packets to the victim node.

In order to mitigate the aforementioned security implications, appropriate packet filtering policies should be enforced. In general transit routers should be more permissive in terms of the traffic they allow (and hence employ a black-list approach to packet filtering), whilst nodes closer to the edge of the network (e.g., enterprise border routers) should generally be more conservative and only allow traffic they are expecting to receive (i.e. employ a white-list

approach to packet filtering). However, we note that the packet filtering policy is likely to depend on a number of operational factors (see e.g. [IPV6-EHS]) and the capabilities and performance properties (see e.g. [IPV6-FW]) of network devices.

Some networks have resorted to filtering packets that employ extension headers, affecting the reliability of IPv6 extension headers when they are employed on the public Internet [RFC7872]. Measurements carried out during the publication process of [RFC7872] (but somehow not included in that document) indicate that the widespread practice of dropping IPv6 packets that contain extension headers also affects IPsec extension headers. The unfortunate consequences of this are that it may be necessary to tunnel IPsec traffic over some transport protocol (e.g. TCP or UDP) for the IPsec packets to survive the public IPv6 Internet. For some use cases, alternative technologies such as TLS VPNs might be employed instead.

2.3. Fragmentation

In contrast with the IPv4 world, where fragmentation can be performed both by the sending hosts and by intermediate routers, IPv6 fragmentation is performed only by hosts. This relieves IPv6 routers from the expensive task of fragmenting packets.

An important aspect of IPv6 fragmentation is that support for fragmentation is implemented by means of IPv6 extension headers (specifically, the Fragment Header). Former specifications of the IPv6 protocols (i.e., those that predated [RFC8200]) allowed some pathological fragmentation cases, such as where the first fragment of a packet does not contain the entire IPv6 header chain (see [RFC7112]).
Such pathological fragmentation cases may still be allowed by legacy IPv6 implementations, and thus might be leveraged to circumvent IPv6 security controls [IPV6-FW-2]. Additionally, since fragmentation support is implemented by means of IPv6 extension headers, all general security considerations for extension headers apply for the Fragment Header.

We note that whilst support for IPv6 fragmentation is required in a number of scenarios (including DNS traffic and IPv6-based tunnels), recent research in this area indicates there is widespread filtering of IPv6 fragments in the public IPv6 Internet (see [RFC7872] and [IPV6-FRAG]). Recent work at the IETF deprecates the use of fragmentation in the public Internet [IPV6-FRAG-2].

2.4. IPsec support

A common myth associated with IPv6 is the expectation of increased usage of IPsec. If anything, such expectation may be based on the fact that legacy IPv6 specifications (namely [RFC4294]) originally required the implementation of IPsec by all IPv6 nodes. However, such requirement never resulted into more widespread implementation of IPsec (and even less into increased usage of IPsec) and was eventually removed in a subsequent revision of the specification ([RFC6434], now superseded by [RFC8504]).

On the other hand, that NAT devices no longer need to be deployed in IPv6 networks, has been seen as an opportunity for increased usage of native IPsec traffic (as opposed to tunneled IPsec traffic). Unfortunately, unpublished measurements (carried out as part of the work behind [RFC7872]) seem to indicate that IPv6 packets employing IPsec extension headers suffer from widespread packet drops, in the same way as other types of IPv6 extension headers. Therefore, it may be necessary to encapsulate IPsec traffic in other protocols (such as UDP [RFC3948]) for IPsec to become usable across the public IPv6 Internet.

2.5. Fault Isolation

Similar to its IPv4 counterpart, IPv6 employs ICMPv6 [RFC4443] for fault isolation. For the most part, most ICMPv6 error messages are similar to the ICMP messages from the IPv4 world.

It is important to consider a number of differences between ICMPv6 and ICMPv4:

- Most popular implementations of connection-oriented transport protocols such as TCP do not abort connections upon receipt of ICMPv6 error messages, and thus are not vulnerable to the connection-reset attacks described in [RFC5927].

- Many IPv6 implementations fail to perform basic validation checks on incoming ICMPv6 error messages. Some implementations can be easily fooled to accept ICMPv6 "Packet Too Big" error messages (claiming an MTU smaller than 1280) and generate IPv6 fragments that can lead to DoS conditions. Please see [RFC8021], [IPV6-AT], and [IPV6-AT-2] for details.

2.6. Address Resolution

IPv6 implements address resolution by means of ICMPv6 Neighbor Solicitation (NS) and ICMPv6 Neighbor Advertisement (NA) messages. These messages are analogous to ARP Request and ARP Reply messages [RFC826], respectively. Similarly, an IPv6 data structure called "Neighbor Cache" records the mappings between IPv6 addresses and link-layer addresses, along with information about the "freshness" of such mapping.

Attacks against the address resolution mechanism not only include the typical "man in the middle" and "Denial of Service" equivalents of the IPv4 world, but also include "Neighbor Cache Exhaustion" (NCE) attacks which typically crash the victim system as a result of too many bogus entries in the Neighbor Cache. The aforementioned situation may be triggered by a deliberate NCE attack, or as a side-effect of an address scan in which the last router to the target network is unable to cope with an unusually large number of entries in its Neighbor Cache (one for each IPv6 address that has been probed on the target network). Whilst the problem should be properly addressed via careful implementation of the address resolution mechanism, a number of operational mitigations are also available [ND-INDEF].

One subtle, but very important difference between address resolution in IPv6 and IPv4 is that NS and NA messages are ICMPv6 messages encapsulated in IPv6 packets, and therefore they could, in theory, employ IPv6 extension headers (including Fragment Headers). This leads to more complex traffic that can be difficult to policy, e.g. at layer-2 devices such as switches.

Whilst use of fragmentation with Neighbor Discovery has been recently deprecated (see [RFC6980] and [RFC8200]), legacy systems might still accept such packets and thus it is not possible to rely on all nodes to drop them. We note that use of other IPv6 extension headers is still allowed, even when they could also be challenging to devices that must inspect and/or policy Neighbor Discovery traffic (e.g. at layer 2).

When it comes to address resolution attacks (excluding the NCE attacks discussed above), the following mitigations are (theoretically) available:

- Secure Neighbor Discovery (SEND)
- Network monitoring
- Network compartmentalization
- Enforcing packet filtering at layer-2 devices

The following subsections discuss each of these techniques.

2.6.1. Secure Neighbor Discovery (SEND)

SEND is specified in [RFC3971] and is usually deemed (both in IPv6 literature and in some IETF specifications) as the final solution to Neighbor Discovery attacks. SEND employs:

- Cryptographically-Generated Addresses (CGA) to bind IPv6 addresses to an asymmetric key pair
- RSA signatures to protect all Neighbor Discovery messages
- Certification paths to certify the authority of routers

However, SEND is virtually impossible to deploy in any real-world network scenario because:

- There is virtually no support for SEND in any major operating system (such as MS Windows, Ubuntu, Mac OS, Android, or FreeBSD)
- The requirement of a Public Key Infrastructure (PKI) presents a major obstacle for its deployment

- In most network scenarios, the benefits of deploying SEND do not justify the major efforts that would be required to deploy it, particularly when other internet technologies still remain to be secured (DNS, etc.)

2.6.2. Traffic Monitoring

Whilst network monitoring cannot really mitigate attacks against IPv6 address resolution, it may at least detect and signal the occurrence of such attacks. Monitoring may be performed with general-purpose Network Intrusion Detection Systems (NIDS), or with special-purpose tools such as NDPMon [NDPMON].

2.6.3. Traffic Compartmentalization

Segmenting a network into multiple broadcast domains limits the ability of nodes of attacking other nodes. Whilst it does not eliminate the ability of the attacker of attacking on the same broadcast domain, it does limit the possible impact of the attack.

2.6.4. Enforcing Packet-filtering at Layer-2 Devices

Some Layer 2 devices may offer features to inspect and policy Neighbor Discovery traffic (including traffic address resolution traffic). These devices assume ownership of addresses on a First Come First Served (FCFS) basis: when a Layer 2 device first learns about the mapping of an IPv6 address to a link-layer address, the mapping is assumed to be legitimate. Any subsequent traffic that would override such mapping will be dropped by the Layer 2 device. This is normally referred to as "Neighbor Discovery Inspection" by some vendors of Layer 2 devices. It is important to note that many real-world implementations of this mechanism are subject to evasion attacks by means of IPv6 extension headers. Operators are therefore urged to evaluate the status of their implementations before they are relied upon.

2.7. Address generation/configuration

IPv6 specifies two different mechanisms for automatic host configuration:

- Stateless Address Auto-Configuration (SLAAC)
- Dynamic Host Configuration Protocol for IPv6 (DHCPv6)

SLAAC is mandatory, whilst DHCPv6 is optional [RFC8504]. In theory, Router Advertisement (RA) messages signal which features of these two protocols should be employed, by means of the "M" ("Managed address configuration") and "O" ("Other configuration") bits. However, the interaction between SLAAC and DHCPv6 is far from simple, and there may be different configuration outcomes depending on which information is made available with these two protocols: the operating systems in question, timing parameters, and other aspects (see [SLAAC-P] for details).

As a result of the complex interaction between these two protocols, networks meaning to mitigate automatic configuration attacks should mitigate both SLAAC-based and DHCPv6-based attacks, regardless of which specific protocol is employed for automatic configuration on the local network.

Mitigations for attacks against automatic host configuration theoretically include:

- Secure Neighbor Discovery (SEND)
- Network monitoring
- Network compartmentalization

- Enforcing packet filtering at Layer 2 devices

SEND, network monitoring, and network compartmentalization, are essentially the same mitigations as those for address resolution attacks, and thus the same considerations apply.

Similarly to the "DHCP-snooping" mechanism from the IPv4 world, some Layer 2 devices may be able to enforce packet filtering policies such that incoming Router Advertisements (RAs) and DHCPv6-server packets are allowed only on specific ports of a Layer 2 device (that have been explicitly configured by the network administrator for such purpose). If RA or DHCPv6-server packets are received on any other port, they are silently dropped. RA-Guard [RFC6104][RFC6105] is an implementation of such packet-filtering policy for Router Advertisements messages, whilst DHCPv6-Shield [RFC7610] (sometimes also known as DHCPv6-Guard or DHCPv6-snooping) mitigates attacks based on DHCPv6-server packets.

It should be noted that many real-world implementations of these mechanisms are subject to evasion attacks by means of IPv6 extension headers (see e.g. [RFC7113]). Therefore, operators should evaluate the status of the implementations they employ before they can be relied upon.

2.8. Multicast Usage

Address resolution and SLAAC employ multicast addresses for the IPv6 Destination Address of some of the associated packets. Conceptually speaking, nodes wishing to receive multicast, must join the corresponding multicast group which implies using the Multicast Listener Discovery (MLD) protocol.

MLD is a general protocol employed for "real" multicast traffic that spans across multiple subnets, allowing nodes to learn on which network segments they must forward IPv6 packets destined to specific multicast addresses. However, when it comes to link-local multicast traffic, the only usage of MLD is to allow MLD-snooping switches to inspect MLD packets to learn on which switch ports multicasted packets should be retransmitted.

There are two versions of MLD: MLD [RFC2710] and MLDv2 [RFC3810]. MLDv2 adds the ability for a node to report interest in packets destined to a particular multicast address only from specific source addresses or from all sources except for specific source addresses. This can be very useful functionality for general multicast traffic, but not for multicast traffic associated with address resolution or automatic configuration. Indeed, the increased complexity of MLDv2 is rather

overkill and unwarranted, soin scenarios where the only use case of multicast is that associated with Address Resolution and SLAAC, it may be desirable to employ MLD over MLDv2.

[MLD-SEC] discusses some possible attacks against Multicast Listener Discovery.

2.9. Network Architecture

The most common architecture of IPv4 networks consists of internal nodes employing private IPv4 addresses space [RFC1918], connected to the external/public network via a NAT device. As a side-effect of translating IPv4 addresses and transport-protocol port numbers, NAT devices end up enforcing a filtering policy of "only allowing outgoing communications".

Whilst this is certainly not a security panacea, it does reduce the attack surface in many network scenarios [FIREWALLS] [IPV6-IOT]. Since IPv6 networks need not rely on NAT devices, it is sometimes assumed that IPv6 nodes will be subject to increased exposure - that is, that each IPv6 node will be directly reachable from the public Internet. However, this need not, and generally should not be the case.

For example, a network that currently employs IPv4 private address space and connects to the public Internet via a NAT device. may limit IPv6 host exposure by deploying a stateful IPv6 firewall at the same point of the network topology where the IPv4 NAT device is located. Such IPv6 firewall would normally be configured to "only allow outgoing communications", such that the IPv6 filtering policy parallels its IPv4 counterpart. Additionally, IPv6 hosts may employ host-based IPv6 firewalls that "only allow outgoing communications", in the same way that many IPv4 hosts do for IPv4 traffic.

Such IPv6 network "architecture" and packet-filtering policy is one of the allowed default settings in the IETF recommendations for Customer Premises Equipment (CPE) for providing residential IPv6 Internet Service [RFC6092] and is commonly seen in emerging IPv6 deployments. Ironically, some of these networks lack support for helper protocols (such us UPnP) that enable operation of peer-to-peer (P2P) applications [IPv6-P2P] across these types of middleboxes.

3. Security Implications of Dual-Stack Networks

The IPv6 protocol suite "simply" provides an alternative network-layer service to the upper-layer protocols. Thus, dual-stack servers will normally offer the same network services over both

Internet protocols. After all, network services tend to be agnostic with respect to the underlying Internet protocol.

From a security standpoint, it is important that the same security policies are enforced on both Internet protocols, since otherwise attackers would simply employ the protocol that represents less resistance to attacks: whether that's penetrating a network server, or performing a Denial of Service attack against a host or network, etc.

One of the most simple and widespread security controls is the enforcement of packet filtering policies via some form of firewall device. Whilst IPv6 packet filtering policies have traditionally been assumed to be "weaker" than their IPv4 counterparts (probably as a result of limited experience with the IPv6 protocols and/or limited IPv6 support in network security devices), recent research [IPV6-POL] suggests that whilst mismatches between the filtering policies for both protocols are common, there is no clear indication of the policies for one protocol being weaker than those for the other.

It is important to stress that for most network scenarios, the security policies enforced for IPv6 should be equivalent to those enforced for IPv4. However, we acknowledge that limited support in security devices (whether in terms of features or in terms of performance) may hinder the achievement of that goal [IPV6-FW] [IPV6-FW-2].

4. Security Implications of IPv6 on IPv4 Networks

Whenever a dual-stack host intends to connect to another host, it will typically employ the DNS to obtain IPv4 and IPv6 addresses for the target host's domain name. Subsequently, it will try to communicate with the aforementioned host by trying either each address in sequence or some pair of the addresses in parallel (see e.g. [RFC6555] and [RFC8305]).

Normally, a host on an IPv4-only network will not configure IPv6 global unicast addresses or IPv6 default routes, and hence communication attempts employing IPv6 (if any) will fail with only IPv4 having a chance to succeed.

Most modern operating systems support IPv6 and have such support enabled by default, regardless of whether IPv6 has been deployed on the network to which the nodes are attached. This means that even if global IPv6 connectivity is missing, impending IPv6 connectivity is nevertheless present in the otherwise IPv4-only network. In other words, most "IPv4-only

networks" are composed of dual-stack nodes that could readily leverage IPv6 connectivity when/if it becomes available.

Thus, an attacker connected to the local subnet could trigger IPv6 network configuration (e.g. by sending forged Router Advertisement messages) and subsequently perform IPv6-based attacks such as Denial of Service (DoS), Man In The Middle (MITM), triggering VPN traffic leakages, or simply causing traffic to employ IPv6 (e.g. if the attacker assumes there are fewer or no security controls for the IPv6 case). Some security issues, such as Layer 3 Virtual Private Network (VPN) Tunnel Traffic Leakages, might even take place inadvertently when a node employing an IPv4-only VPN tunnel client connects to a dual-stack network.

As a result, even networks that are meant to be IPv4-only should enforce IPv6 security controls – if only to make sure that the network really supports only IPv4 (and not IPv6). These controls may range from mitigating attacks against automatic configuration and address resolution mechanisms, to enforcing IPv6 ACLs or blocking IPv6 traffic altogether at Layer 2.

It is important to note that a number of transition/co-existence mechanisms may provide IPv6 connectivity by tunneling IPv6 packets over other protocols. In such cases, IPv6 security controls should be enforced on the tunnel payload – a feature that might or might not be readily available. Since it is expected that most organizations deploy IPv6 in the short or near term, enforcing IPv6 security controls is generally preferable over simply disabling IPv6 support in all nodes. However, in IPv4-only network scenarios where enforcing IPv6 security controls is not feasible, networks may have to resort to block IPv6 traffic at Layer 2 devices in order to mitigate IPv6 attacks against IPv4 networks.

[RFC7123] elaborates on IPv6 security attacks against IPv4 networks and discusses possible mitigation techniques. [RFC7359] discusses Layer 3 Virtual Private Network (VPN) Tunnel Traffic Leakages and discusses possible mitigations.

5. Acknowledgements

Kevin Meynell and Jan Žorž provided valuable comments on earlier versions of this document.

6. References

[BROERSMA] Broersma, R., "IPv6 Everywhere: Living with a Fully IPv6-enabled environment", Australian IPv6 Summit 2010, Melbourne, VIC Australia, October 2010. http://www.ipv6.org.au/10ipv6summit/talks/Ron_Broersma.pdf

[CTF] Gont, F. "Network reconnaissance: How to use SI6 Networks' IPv6 toolkit". TechTarget article, August 2018. https://searchsecurity.techtarget.com/tip/Network-reconnaissance-How-to-use-SI6-Networks-IPv6-toolkit

[DNS-REV] Gont, F. "How to use DNS reverse mapping to scan IPv6 addresses", TechTarget article, February 2017. https://searchsecurity.techtarget.com/tip/How-to-use-DNS-reverse-mapping-to-scan-IPv6-addresses

[DNS-REV2] Gont, F., "DNS reverse address mapping: Exploiting the scanning technique", TechTarget article, February 2017. https://searchsecurity.techtarget.com/tip/DNS-reverse-address-mapping-Exploiting-the-scanning-technique

[FIREWALLS] Gont, F., Baker, F., "On Firewalls in Network Security", IETF Internet-Draft (draft-gont-opsawg-firewalls-analysis), work in progress. https://tools.ietf.org/html/draft-gont-opsawg-firewalls-analysis

[IPV6-ADDR] Gont, F., Gont, G., Garcia Corbo, M., Huitema, C., "Problem Statement Regarding IPv6 Address Usage", IETF Internet-Draft (draft-gont-6man-address-usage-recommendations), work in progress. https://tools.ietf.org/html/draft-gont-6man-address-usage-recommendations

[IPV6-AT] Gont, F., "How a single ICMPv6 packet can cause a denial-of-service attack", TechTarget article, March 2017. https://searchsecurity.techtarget.com/tip/How-a-single-ICMPv6-packet-can-cause-a-denial-of-service-attack

[IPV6-AT-2] Gont, F., "Using IPv6 atomic fragments for a denial-of-service attack", TechTarget article, March 2017. https://searchsecurity.techtarget.com/tip/Using-IPv6-atomic-fragments-for-a-denial-of-service-attack

[IPV6-EHS] Gont, F., Hilliard, N., Doering, G., Liu, W., Kumari, W. "Operational Implications of IPv6 Packets with Extension Headers", IETF Internet-Draft (draft-gont-v6ops-ipv6-ehs-packet-drops), work in progress. https://tools.ietf.org/html/draft-gont-v6ops-ipv6-ehs-packet-drops

[IPV6-FHS] Gont, F., "First-hop security in IPv6", TechTarget article, January 2012. https://searchenterprisewan.techtarget.com/tip/First-hop-security-in-IPv6

[IPV6-FRAG] Huston, G., "Surviving IPv6 Fragmentation", October 2017. https://labs.apnic.net/presentations/store/2017-10-25-xtn-hdrs-dns.pdf

[IPV6-FRAG-2] Bonica, R., Baker, F., Huston, G., Hinden, B., Troan, O., Gont, F. "IP Fragmentation Considered Fragile", IETF Internet-Draft (draft-ietf-intarea-frag-fragile), work in progress. https://tools.ietf.org/html/draft-ietf-intarea-frag-fragile

[IPV6-FW] Zack, E., "IPv6 Security Assessment and Benchmarking", IPv6 Hackers Meeting #1, Berlin, Germany, July 2013. http://www.ipv6hackers.org/meetings/ipv6-hackers-1

[IPV6-FW-2] Gont, F., "IPv6 firewall security: Fixing issues introduced by the new protocol", TechTarget article, November 2011. https://searchenterprisewan.techtarget.com/tip/IPv6-firewall-security-Fixing-issues-introduced-by-the-new-protocol

[IPV6-MYTHS] Gont, F. "IPv6 myths: Debunking misconceptions regarding IPv6 security features". TechTarget article, May 2011. https://searchsecurity.techtarget.com/tip/IPv6-myths-Debunking-misconceptions-regarding-IPv6-security-features

[IPV6-IOT] Gont, F., "How IPv6 deployment affects the security of IoT devices", TechTarget Article, October 2017. https://internetofthingsagenda.techtarget.com/feature/How-IPv6-deployment-affects-the-security-of-IoT-devices

[IPV6-P2P] Gont, F., "Ensuring P2P apps don't cause network performance issues with IPv6", TechTarget Article, September 2018. https://searchnetworking.techtarget.com/tip/Ensuring-P2P-apps-dont-cause-network-performance-issues-with-IPv6

[IPV6-POL] Gont, F., "What to do when IPv4 and IPv6 policies disagree", TechTarget article, August 2018. https://searchsecurity.techtarget.com/tip/What-to-do-when-IPV4-and-IPv6-policies-disagree

[IPV6-REC] Gont, F., "How to perform IPv6 network reconnaissance", TechTarget article, July 2015. https://searchsecurity.techtarget.com/tip/How-to-perform-IPv6-network-reconnaissance

[IPV6-SEC] Gont, F., "Address IPv6 security before your time runs out", TechTarget article, March 2013. https://searchsecurity.techtarget.com/feature/Address-IPv6-security-before-your-time-runs-out

[IPV6-SEC-2] Pepelnjak, I., "IPv6 security issues: Fixing implementation problems", TechTarget article, January 2011. https://searchtelecom.techtarget.com/tip/IPv6-security-issues-Fixing-implementation-problems

[MLD-SEC] Atlasis, A., Salazar, J., "MLD Considered Harmful – Breaking Another IPv6 Subprotocol". Troopers 2015 Conference, Heidelberg, Germany, 2015. https://www.troopers.de/media/filer_public/7c/35/7c35967a-d0d4-46fb-8a3b-4c16df37ce59/troopers15_ipv6secsummit_atlasis_rey_salazar_mld_considered_harmful_final.pdf

[ND-ATTCK] Gont, F. "How to protect your IPv6 address management", TechTarget article, January 2015.
https://searchnetworking.techtarget.com/tip/How-to-protect-your-IPv6-address-management

[ND-INDEF] Jaeggli, J., “Indefensible Neighbors”, IEPG Meeting - July 2018 @ IETF 102.
http://www.iepg.org/2018-07-15-ietf102/indefensible-neighbors.pdf

[NDPMON] Beck, F., Cholez, T., Festor, 0., Chrisment, I., "Monitoring the Neighbor Discovery Protocol". The Second International Workshop on IPv6 Today - Technology and Deployment - IPv6TD 2007, Mar 2007, Guadeloupe/French Caribbean, Guadeloupe.
https://hal.inria.fr/inria-00153558/document

[OPSEC-ND] Gont, F., Bonica, R., Liu, W., "Security Assessment of Neighbor Discovery (ND) for IPv6", IETF Internet-Draft (draft-ietf-opsec-ipv6-nd-security), work in progress.
https://tools.ietf.org/html/draft-ietf-opsec-ipv6-nd-security

[OPSEC-V6] Vyncke, E., Chittimaneni, K., Kaeo, M., Rey, E., "Operational Security Considerations for IPv6 Networks", IETF Internet-Draft (draft-ietf-opsec-v6), work in progress.
https://tools.ietf.org/html/draft-ietf-opsec-v6

[RFC826] 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

[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

[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

[RFC2710] Deering, S., Fenner, W., and B. Haberman, "Multicast Listener Discovery (MLD) for IPv6", RFC 2710, DOI 10.17487/RFC2710, October 1999. https://www.rfc-editor.org/info/rfc2710

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

[RFC3948] Huttunen, A., Swander, B., Volpe, V., DiBurro, L., and M. Stenberg, "UDP Encapsulation of IPsec ESP Packets", RFC 3948, DOI 10.17487/RFC3948, January 2005. http://www.rfc-editor.org/info/rfc3948

[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

[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

[RFC4291] Hinden, R. and S. Deering, -2IP Version 6 Addressing Architecture", RFC 4291, DOI 10.17487/RFC4291, February 2006. https://www.rfc-editor.org/info/rfc4291

[RFC4294] Loughney, J., Ed., "Ipv6 Node Requirements", RFC 4294, April 2006. https://www.rfc-editor.org/info/rfc4294

[RFC4692] Huston, G., "Considerations on the IPv6 Host Density Metric", RFC 4692, October 2006. https://www.rfc-editor.org/info/rfc4692

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

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

[RFC4941] Narten, T., Draves, R., and S. Krishnan, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", RFC 4941, DOI 10.17487/RFC4941, September 2007. https://www.rfc-editor.org/info/rfc4941

[RFC5927] Gont, F., "ICMP Attacks against TCP", RFC 5927, DOI 10.17487/RFC5927, July 2010. https://www.rfc-editor.org/info/rfc5927

[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, February 2011. http://www.rfc-editor.org/info/rfc6104

[RFC6105] Levy-Abegnoli, E., Van de Velde, G., Popoviciu, C., Mohacsi, J., "IPv6 Router Advertisement Guard", RFC 6105, February 2011. http://www.rfc-editor.org/info/rfc6105

[RFC6164] Kohno, M., Nitzan, B., Bush, R., Matsuzaki, Y., Colitti, L., and T. Narten, "Using 127-Bit IPv6 Prefixes on Inter-Router Links", IETF RFC 6164, April 2011. http://www.rfc-editor.org/info/rfc6164

[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

[RFC6555] Wing, D. and A. Yourtchenko, "Happy Eyeballs: Success with Dual-Stack Hosts", RFC 6555, DOI 10.17487/RFC6555, April 2012.
https://www.rfc-editor.org/info/rfc6555

[RFC6583] Gashinsky, I., Jaeggli, J., Kumari, W. "Operational Neighbor Discovery Problems", RFC 6583, March 2012.
http://www.rfc-editor.org/info/rfc6583

[RFC6583] Gashinsky, I., Jaeggli, J., Kumari, W. "Operational Neighbor Discovery Problems", RFC 6583, March 2012.
http://www.rfc-editor.org/info/rfc6583

[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.
http://www.rfc-editor.org/info/rfc7113

[RFC7123] Gont, F. and W. Liu, "Security Implications of IPv6 on IPv4 Networks", RFC 7123, February 2014.
http://www.rfc-editor.org/info/rfc7123

[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.
http://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, August 2014.
http://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

[RFC7421] Carpenter, B., Ed., Chown, T., Gont, F., Jiang, S., Petrescu, A., and A. Yourtchenko, "Analysis of the 64-bit Boundary in IPv6 Addressing", RFC 7421, DOI 10.17487/RFC7421, January 2015.
https://www.rfc-editor.org/info/rfc7421

[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

[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

[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

[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

[RFC8021] Gont, F., Liu, W., and T. Anderson, "Generation of IPv6 Atomic Fragments Considered Harmful", RFC 8021, DOI 10.17487/RFC8021, January 2017.
https://www.rfc-editor.org/info/rfc8021

[RFC8064] Gont, F., Cooper, A., Thaler, D., and W. Liu, "Recommendation on Stable IPv6 Interface Identifiers", RFC 8064, February 2017.
https://www.rfc-editor.org/info/rfc8064

[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

[RFC8305] Schinazi, D. and T. Pauly, "Happy Eyeballs Version 2: Better Connectivity Using Concurrency", RFC 8305, DOI 10.17487/RFC8305, December 2017. https://www.rfc-editor.org/info/rfc8305

[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

[RTR-ARCH] Petersen, B. and J. Scudder, "Modern Router Architecture for Protocol Designers", IEPG 94. Yokohama, Japan. November 1, 2015. http://www.iepg.org/2015-11-01-ietf94/IEPG-RouterArchitecture-jgs.pdf

[SI6-FRAG] Gont, F., "IPv6 NIDS evasion and improvements in IPv6 fragmentation/reassembly", 2012. http://blog.si6networks.com/2012/02/ipv6-nids-evasion-and-improvements-in.html

[SLAAC-P] Liu, B., Jiang, S., Gong, X., Wang, W., Rey, E., "DHCPv6/SLAAC Interaction Problems on Address and DNS Configuration", IETF Internet-Draft (draft-ietf-v6ops-dhcpv6-slaac-problem), work in progress. https://tools.ietf.org/html/draft-ietf-v6ops-dhcpv6-slaac-problem