University of West Bohemia
Faculty of Applied Sciences
Department of Computer Science and Engineering

Master Thesis

Vulnerabilities and Threats in IPv6 Environment

Pilsen 2013

Petr Fojtů

Solemn Declaration

I hereby declare that this master thesis is completely my own work and that I used only the cited sources.

Pilsen; June 25, 2013

Petr Fojtů

Abstract

Vulnerabilities and Threats in IPv6 Environment This thesis reviews IPv6 security with focus on Local Area Networks and IDS/IPS systems. It compares IPv4 and IPv6 threats, vulnerabilities and gives basic security recommendations. Selected IPv6 attacks and exploits are demonstrated in simulated attacker/victim scenario on IPv6 network. These experiments are then used to set up guidelines for evaluating usability of IDS/IPS appliances against IPv6-specific threats.

Disclaimer

The testing was performed solely for the purpose of applying engineering practices in order to meet the requirements necessary for successful elaboration of this thesis. Due to the fact that the testing was not conducted according to methodology recognized by International Business Machines Corp. (refered to as *company* hereafter) and was not performed in a laboratory certified for this purpose, the results do not necessarily have to reflect actual performance of the appliance as it may be when deployed in real environment. Therefore neither the company nor IBM Česká republika, spol. s r.o. can guarantee reliability of the results. The company, IBM Česká republika spol. s r.o. and author of this work claim absolutely no responsibility for any potential misuse of this work for illegal purposes.

Contents

List of Figures

List of Tables

1 Preface

The Internet today is in a strange situation. Global pool of available IPv4 address has been already depleted and IPv6 is has not been widely utilized yet. There can be rather a long discussion about the cause. Seems like the popular IT phrase *"If it works, don't fix it."* applies once again. This, however, is not a sustainable state. It is already late and IPv6 deployment growth will have to hurry up to catch up with current Internet growth.

One of the obstacles to be overcome is the fear of unknown. IPv4 has been around for years and best practices are well proven. The biggest challenge of the Internet today is the one of security. When IPv4 was designed, security was not a major concern. The current security requirements were known to IPv6 designers and IPv6 is hopefully expected to fulfill them.

This works could help to answer the importat question whether IPv6 is more secure than IPv4. It is intended to gather current knowledge regarding the security of IPv6. It shall build on experience gained through IPv4 and highlight the differences. Threats and known vulnerabilities of the protocol will be discussed and examined as well as appropriate basic countermeasures to address them. The focus will be intrusion detection/intrusion prevention systems and local area networks.

Another common concern is that IPv6 implementations are not mature enough. List of *Commom Vulnerabilities and Exposures* (CVE) associated with IPv6 sides the argument. As IPv6 is not widely used, it is not widely tested either. Practical part shall zero in on particular vulnerabilities and available testing tools. The output shall be a know-how how to utilize these tools in the name of improving security. Testing will be performed on available IDS/IPS solution. Nevertheless, the guidelines shall be easily adjustable so such testing can be performed on almost any security system.

2 Internet Protocol version 6

The reason behind *Internet Protocol version 6* (IPv6) development is to address shortcomings of its predecessor, *Internet Protocol version 4* (IPv4), maninly the size of its address space. New features needed in the modern world such as mobility and security are introduced on this occasion. IPv6 has been developed for a rather long period of time. *Internet Engineering Task Force* (IETF) recommended IPv6 in [*RFC* 1752], published in January 1995, thus there have been rather big expectations from IPv6 over the years.

Nevertheless, it should be always taken into account that IPv6 introduces changes at the third layer of the ISO/OSI model (outlined in Figure 2.1) while other layers are mostly unaffected. Possible drawbacks of other layers are still present and remain intact.

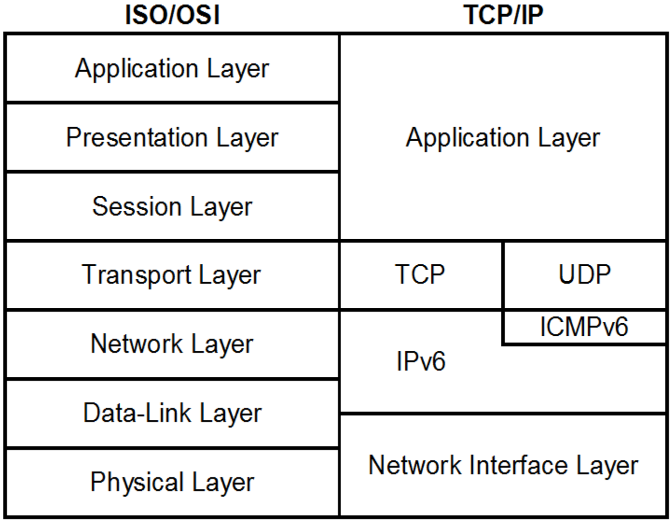


Figure 2.1: ISO/OSI Model and TCP/IP Stack

Layer 3 of the ISO/OSI model, *Network Layer*, provides logical addressing which is used by routers for path determination. It is responsible for packet forwarding and data transfers among hosts on different networks.

2.1 IPv6 Features and Benefits

The most significant differences between IPv4 and IPv6 can be listed and briefly described as follows:

Adress space and addressing Main reason for IPv6 deployment - IP address is 128 bits long, instead of 32 bits. This should provide enough addresses (up to 2^{128}) for forseeable future. Addresses are usually written in hexadecimal notation, see Figure 2.2 for an example. *Multicast* addresses designed for efficient one-to-many communicaton and *anycast* for redundant services (also known as one-to-one-of-many) are introduced. On the contrary, *broadcasts* are not implemented.

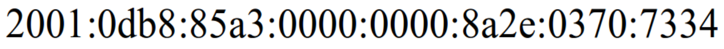

2001:0db8:85a3:0000:0000:8a2e:0370:7334

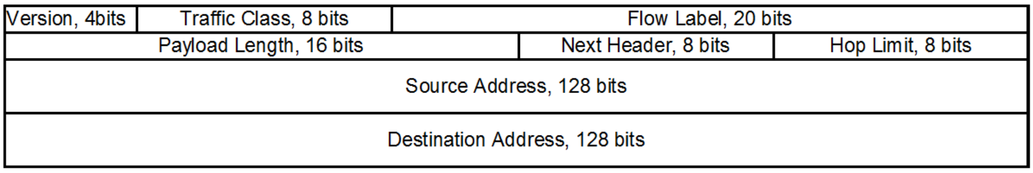
Figure 2.2: Example of an IPv6 Address

Route Aggregation IPv6 addresses should be assigned hierarchically. The structure then provides for simple summarization and consequently for lighter exchange of routing information on the Internet. The large address space allows organizations to obtain continous blocks of addresses, which should be assigned by Internet service providers.

Autoconfiguration Basically a plug-and-play networking, providing the ability of *Stateless Addresses Autoconfiguration* (SLAAC) which should occur without the use of *Dynamic Host Configuration Protocol* (DHCP) as defined in [*RFC* 4862].

IPv6 Header New header format is defined in [*RFC* 2460]. In-depth description can be found therein. It has fixed length of 40 bytes and is much simpler. Compared to IPv4 header, fields *IP Header Length*, *Identification*, *Flags*, *Fragment Offset* and *Header Checksum* have been removed. Figure 2.3 illustrates IPv6 packet header format.

With 40 bytes of fixed length and only 8 fields, new header format can improve processing speeds. Every packet has this *base header*, which can be followed by an *extension header* defined in Next Header field. Such chaining is outlined in Figure 2.3. For list of all possible Next Header values please refer to Attachment III.



Version, 4bits	Traffic Class, 8 bits	Flow Label, 20 bits
Payload Length, 16 bits	Next Header, 8 bits	Hop Limit, 8 bits
Source Address, 128 bits		
Destination Address, 128 bits		

Figure 2.3: Format of IPv6 Packet Header

There can be several chained headers in one packet. [*RFC* 2460] defines six types of extension headers: *Hop-by-hop Option Header*, *Routing Header*, *Fragment Header*, *Destination Options Header*, *Authentication Header* (AH) and *Encapsulating Security Payload Header* (ESP). The latter two will be discussed in detail in Sections 3.1 and 3.2, others when needed.

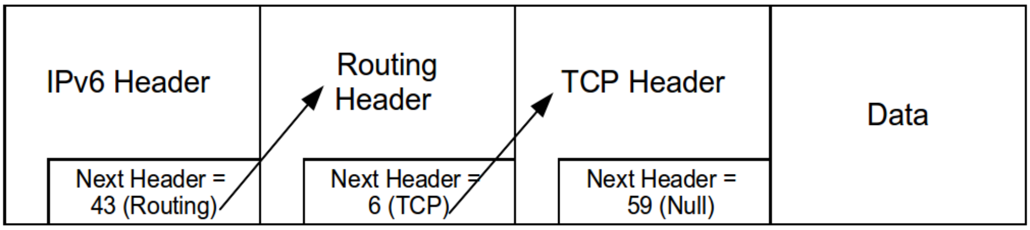


Figure 2.4: IPv6 Extension Headers Chaining Example

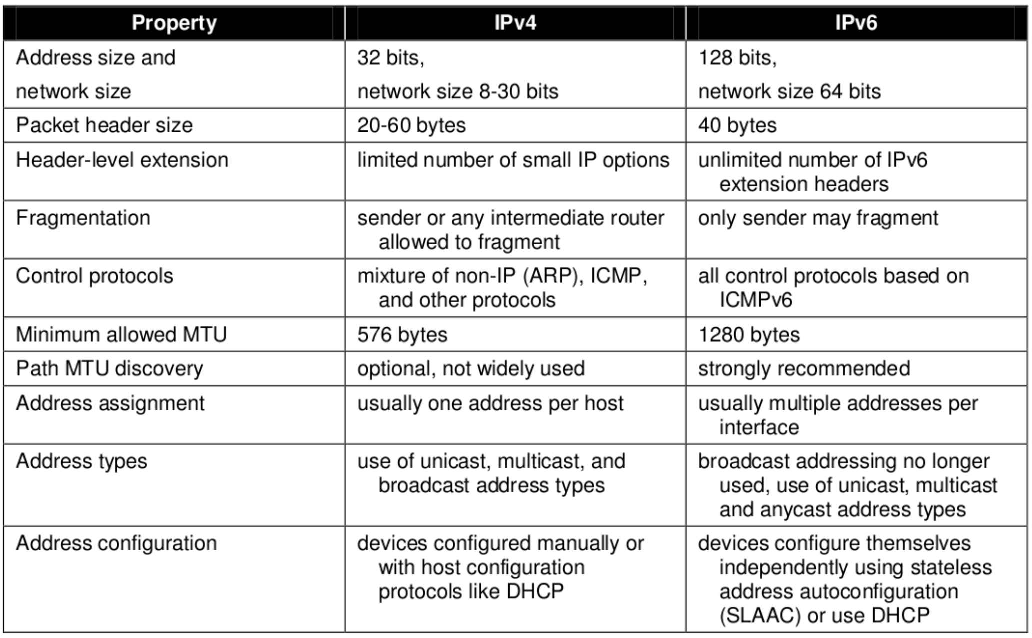
Improved Transmission Fragmentation as known from IPv4 does not actually exist in IPv6. It does not happen at intermediate nodes, packets can be fragmented at source nodes only [*RFC* 2460]. Routers' need for fragmentation is eliminated by mechanism called *Path Maximum Transmission Unit Discovery* (PMTU) defined in [*RFC* 1981]. This mechanism is used by nodes to determine maximum transmission unit size. The source node then uses *Fragment Header* when packet fragmentation is needed.

QoS Support *Quality of Service* (QoS) support is facilitated within the IPv6 packet. Flows can be labeled (using *Traffic Class* and *Flow label* header fields), enabling routers to recognize appropriate flows to which packets belong and making it possible for high priority packets to arrive to their destination in a timely manner. More information can be found in [*RFC* 3697].

Mobility Support *Mobile IPv6* protocol (MIPv6) brings support for moving a node from one network to another wihtout losing connectivity. By retaining its *Home Address* (HoA), nodes can disconnect and reconnect at different place in internet topology (as defined in [*RFC* 6275]). Vast address space is essential for this mechanism.

Native End-to-End Security Unlike IPv4, where support for *Internet Protocol Security* (IPsec) is optional, its implementation in IPv6 is mandated. It provides data integrity by sender authentication and optionally data confidentiality through encryption. In IPv4 world, IPsec typically provides security between border routers (typically for VPN access) due to NAT limitations. There is no need for NAT in IPv6 world, therefore IPsec can be utilized for securing end-to-end communications. However, use of IPsec is not required.

The latter will be discussed in detail in the following chapter as security is crucial for this work. No other explicit security feature has been introduced in IPv6. Implicit security consequences, threat comparison and security considerations of the protocol will be discussed in Chapter 4. Main differences between IPv4 and IPv6 can be summarized as in Figure 2.5.



Property	IPv4	IPv6
Address size and network size	32 bits, network size 8-30 bits	128 bits, network size 64 bits
Packet header size	20-60 bytes	40 bytes
Header-level extension	limited number of small IP options	unlimited number of IPv6 extension headers
Fragmentation	sender or any intermediate router allowed to fragment	only sender may fragment
Control protocols	mixture of non-IP (ARP), ICMP, and other protocols	all control protocols based on ICMPv6
Minimum allowed MTU	576 bytes	1280 bytes
Path MTU discovery	optional, not widely used	strongly recommended
Address assignment	usually one address per host	usually multiple addresses per interface
Address types	use of unicast, multicast, and broadcast address types	broadcast addressing no longer used, use of unicast, multicast and anycast address types
Address configuration	devices configured manually or with host configuration protocols like DHCP	devices configure themselves independently using stateless address autoconfiguration (SLAAC) or use DHCP

Figure 2.5: Summary of Differences Between IPv4 and IPv6 [5]

3 IP Security with IPv6

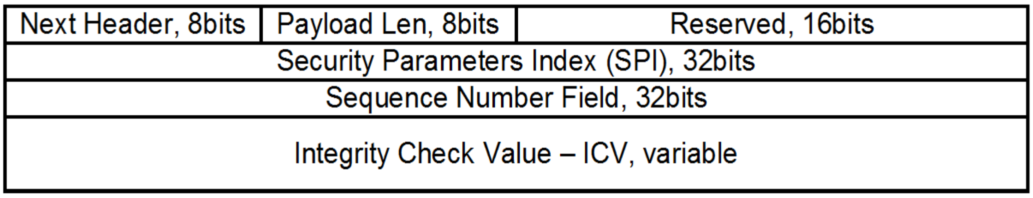
Native IP security support is the most beneficial IPv6 feature from security perspective. IPsec is a security framework defined in [*RFC* 4301]. It secures data on the network level, providing security for upper-layers' data.

Important part of the concept is the idea behind Security Association (SA). SA is a one-way relationship between source and destination, therefore two SAs are necessary for two-way comunnication between hosts. The SA defines IPsec type, mode and all associated parameteres such as encryption algorithm with related keys and so on. Every SA is associated with a policy which determines how the incoming, resp. outgoing packet will be handled. Keys for the communication purposes are negotiated, hosts mutually authenticated and SAs established using Internet Key Exchange protocol, currently version 2 (IKEv2) as defined in [*RFC* 5996].

As it has been already briefly mentioned in the previous chapter, IPsec utilizes two of the newly introduced extension headers - *Authentication Header* and *Encapsulating Security Payload* header. They may be used separately or combined together to provide desired level of security. IPsec can be used in one of two modes, namely *in tunnel mode* or *transport mode.*

3.1 Authentication Header

Authentication Header (AH), as defined in [*RFC* 4302], is designated to provide data integrity, origin authentication and protection against replay attacks. It is specified by value 51 in *Next Header* field of the preceding header. Format of AH is illustrated in Figure 3.1.



Next Header, 8bits	Payload Len, 8bits	Reserved, 16bits
Security Parameters Index (SPI), 32bits		
Sequence Number Field, 32bits		
Integrity Check Value – ICV, variable		

Figure 3.1: Format of Authentication Header

Particular fields in the Authentication Header can be briefly described as follows.

Next Header This field determines the following header. Numbers are registered by IANA and all possible values are listed in Attachment III.

Payload Len AH payload length in 32 bits words minus “2”.

Reserved Reserved bits for future use. Values must be initialized to zero.

Security Parameters Index (SPI) 32-bit value that the target host uses to identify appropriate Security Association.

Sequence Number Field Unsigned 32-bit counter field which is increased by one for every packet. In other words, it is a packet sequence number.

Integrity Check Value (ICV) Value of this field is comupted from immutable headers of the packet, AH and all immutable fields behind it. It must be a multiple of 32 bits and is verifed by target host thus providing data integrity.

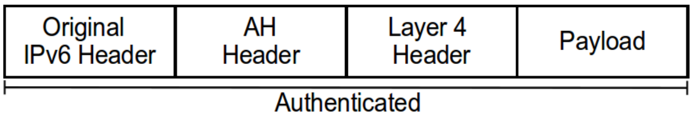


Figure 3.2: Authentication Header in Transport Mode

In *transport mode* (as illustrated in Figure 3.2), original IPv6 header is not altered. Only Authentication Header is added behind it. Where in *tunnel mode* (as illustrated in Figure 3.3), original packet is encapsulated in new, protected packet and Authentication Header is added behind the new IPv6 packet. However, both modes authenticate the whole packet.

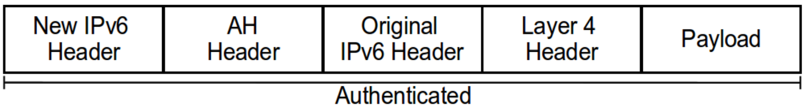


Figure 3.3: Authentication Header in Tunnel Mode

3.2 Encapsulating Security Payload

Encapsulating Security Payload (ESP) header, as defined in [*RFC* 4303], provides data confidentiality, integrity (if required) and limited origin authentication. Its is specified by value 50 in *Next Header* field of the preceding header. Format of ESP is illustrated in Figure 3.4.

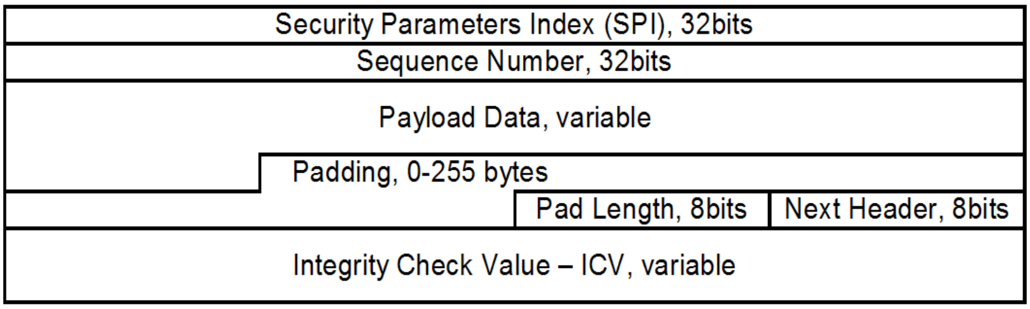


Figure 3.4: Format of Encapsulating Security Payload Header

Particular fields in the Encapsulating Security Payload header can be briefly described as follows.

Security Parameters Index (SPI) 32-bit value that the target host uses to identify appropriate Security Association.

Sequence Number Unsigned 32-bit counter field which is increased by one for every packet. In other words, it is a packet sequence number.

Payload Data This field contains data from the original packet. The data are protected against tampering and disclosure by an encryption.

Padding Bytes added in order to align bytes for encryption purposes.

Pad Length Field indicating the number of padding bytes in the *Padding* field. Valid range is from “0” to “255”.

Next Header This field determines the following header. Numbers are registered by IANA and all possible values are listed in Attachment III.

Integrity Check Value (ICV) Is optional field computed from the ESP header, payload and ESP trailer. It is present only when integrity service is in use. The length depends on algorithm used.

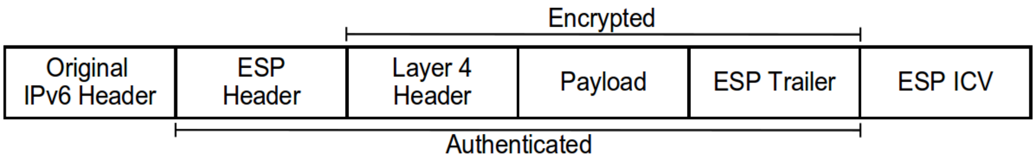


Figure 3.5: Encapsulating Security Payload in Transport Mode

When ESP operates in *transport mode* (as illustrated in Figure 3.5), only Layer 3 payload is protected by encryption, original IPv6 header is left untouched. Operation in *tunnel mode* (as illustrated in Figure 3.6) encapsulates original packet in new, encrypted packet. This mode is considered more secure, because the true sender's and receiver's IP addresses are kept secret. This, however, applies only for communication between security gates and not for end-to-end transmissions. In both cases, significant part of the packet can be authenticated employing EPS ICV trailer.

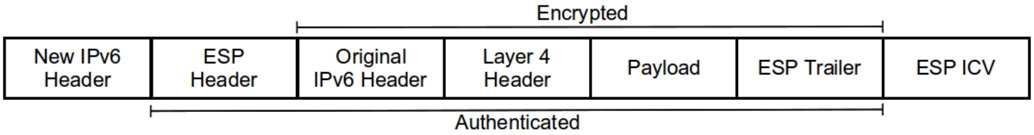


Figure 3.6: Encapsulating Security Payload in Tunnel Mode

4 IPv4 and IPv6 Threat Comparison

It can not be decided whether IPv6 is more secure than IPv4 or not. IPv6 does not introduce a significant improvement apart from prospective IPsec widespread use. The differences between these two protocols are, in most cases, double-edged. Some security threats are very similar or have slightly different considerations; some were mitigated while others were newly introduced.

4.1 Threats with New Considerations

This chapter will discuss threats and attacks with new considerations. It contains those where different approcach is needed due to IPv6 adoption and both the ones where the situation is more complicated for adversary and the ones that endanger the security more.

4.1.1 Reconnaissance

Reconnaissance is first phase of every attack (together with information gathering) and therefore accomplishing good result in this phase is an important building block for subsequent phases. As mentioned at the beginnig of Chapter 2, IPv6 has different address scheme. It implies need for a different approach to reconnaissance.

With 128-bit long addresses and typical subnet prefix of /64 it will be significantly time-consuming to scan the subnets for live hosts. Assuming 10 000 hosts uniformly distributed in such subnet and using traditional brute-force ping sweeps scan, "even at a scan rate of 1 million probes per second (more than 400 Mbps of traffic), it would take more than 28 years of constatnt scanning to find the first active host" [4]. With more typical subnet with 100 hosts, the math is even more interesting, "the number jumps to more than 28 centuries of constant 1-million-packet-per-second scanning to find first host on that first subnet of the victim network" [4]. However, several techniques to speed up this process exist. These are discussed in Chapter 6.

It can be expected that adversary will detour network scanning and focus rather on DNS servers. The servers will be precious source of information. Because IPv6 addresses are generally not easy to remember, dynamic DNS will likely be adopted by administrators. Any patterns or sequences in nodes addressing should be avoided.

4.1.2 Smurf Attacks

Broadcast apmilification attacks, often referred to as *smurf attacks*, are performed by sending ICMP echo request to a broadcast address with spoofed victims address as source address (as illustrated in Figure 4.1). All nodes on the broadcast domain then respond to this request by ICMP echo respond with the formerly spoofed addreess. The victim becomes overwhelmed with traffic causing denial of service situation as a result.

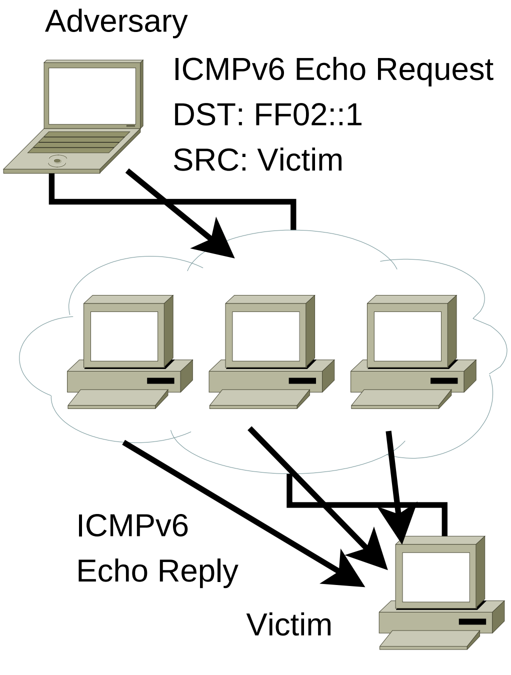


Figure 4.1: Smurf Attack Scheme

This technique is no longer possible with IPv6 because there are no broadcasts. However, multicast addresses can be used instead. Several multicast addresses are currently registered by *Internet Assigned Numbers Authority* (IANA), please refer to Attachment I. Address `FF02:0:0:0:0:0:0:1`, or `FF02::1` for short, represents all nodes on a segment. So it could be a great replacement for broadcast address in IPv4. This was taken into account by IETF and countermeasure is defined in [*RFC* 4443]. ICMP replies should not be generated in response to ICMPv6 messages having a multicast address as a destination. Therefore smurf attack should not be an issue in IPv6 network where all nodes are compliant to [*RFC* 4443].

IPv6 can not function without ICMPv6 as its functionalities are a vital part of the protocol [*RFC* 4443]. Consequently, it can not be completely filtered out like it is often done in IPv4. Attention should be paid to ICMPv6 filtering. IETF defines recommendations for ICMPv6 messages filtering in [*RFC* 4890]. The recommendations are summarized in Attachment II.

4.1.3 Address Spoofing

IP address spoofing is widely utilized by adversary to hide origin of the attack and therefore their identity. The packets are crafted with falsified source IP address, usually from completely different location. [*RFC* 2827] defines protection against spoofing of the network portion of an address. This is done by filtering on the network's edge, where packets with source address outside the valid subnet range are dropped. However, it is not very commonly used countermeasure.

IPv6 Internet should benefit from hierarchical address assignment. Allocations will be easily summarized. As a result, spoofing-preventing filtering should be much easier and in effect more appealing for Internet Service Providers (ISP) to be implemented. But even without spoofing outside of customers address ranges, there is vast range of addresses to be spoofed from within typical IPv6 subnets (2^{64} addresses with */64* prefix).

4.1.4 Routing Security

Corruption of routing information can lead to traffic redirection or connectivity disruption. Exchange of routing information should be well protected. In IPv4, routing protocols are commonly protected using cryptographic authentication. Being extended for IPv6 support, these protocols can be divided into two groups.

First, protocols as *Border Gateway Protocol* (BGP) and *Intermediate System-to-Intermediate System* (IS-IS) did not change their security mechanism with transition to IPv6. BGP authentication uses TCP MD5 signatures based on secret shared by appropriate endpoints [*RFC* 2385]. Similarily, IS-IS exchanges routing information with keyed-hash message authentication code based on MD5 algorithm (HMAC-MD5), which provides integrity and authentication [*RFC* 3567].

Second, *Open Shortest Path First version 3* (OSPFv3) and *Routing Information Protocol Next-Generation* (RIPng) have removed the means of authentication. They both rely on IPsec to provide protection to routing information exchange [2]. IPsec mechanisms are discussed in detail in Chapter 3.

4.1.5 ARP and DHCP Attacks

Address Resolution Protocol (ARP) and *Dynamic Host Configuration Protocol* (DHCP) are the protocols responsible for host initialization in IPv4 networks. Host initialization via DHCP is vulnerable to spoofed responses from rogue DHCP servers. Information obtained from DHCP is IP address, DNS server addres and default gateway. When replaced by adversary, it enables for man-in-the-middle attacks. ARP is used for resolving MAC-IP address pairs. It can be spoofed as well, thus again enabling for MITM attacks.

There is no ARP on IPv6 networks. This functionality was replaced with *Neighbor Discovery Protocol* (NDP) mechanism provided by ICMPv6. Threats endangering NDP are discuseed in Section 4.2.2. Altough the function of DHCP can be partially replaced by stateless address autoconfiguration (SLAAC), it does not provided information like DNS and NTP servers. SLAAC can be complemented or completely replaced by DHCPv6.

DHCPv6 is not an extension of traditional DHCP, it is a new protocol defined in [*RFC* 3315]. But unfortunately, it is vulnerable to very similar threats. It can face starvation or DoS when too many addresses is requested, it has no additional security for preventing rogue devices. When DHCPv6 with sequential allocation is in place, it can spare a lot of adversary's time needed for network scanning.

4.1.6 Internet Worms

Worm is a type of malware designed to exploit a specific vulnerability in a system and then use it to propagate to other systems through the same flaws. Worms may be used to spread virus infection, trojan horses and so on. Worms together with viruses are a significant problem of today's networking.

The basic principles of worms infection does not change with IPv6. It affects ability to propagate of those worms which utilize network scanning to find new targets. The vast and sparsely populated address space of IPv6 will definitely slow down the worm propagation. Other forms of proliferation (through email, instant messaging, peer-to-peer aplication etc.) remain the same. It can be expected that worm developers will focus on these means of propagation or new techniques will be adopted. According to [2], these could be for example: targeting *Domain Name Systems* (DNS) lookups, sniffing neighbor solicitation packets and routing updates or exploit multicast addresses (please refer to Attachment I).

4.2 Newly Introduced Threats

New functionalities always broaden the attack surface available for adversary. As security becomes more and more crucial nowadays, the security aspects should be always considered when designing new feature. To avoid unnecessary risk exposure, unused features and service should be always disabled or handled properly (initialized to zero...).

4.2.1 Extension Header Threats

Extension headers are where all the options from IPv4 packet header were moved to. Extension header is specified in *Next Header* (NH) field of the preceding one. Recommended order of the headers in a packet as per [*RFC* 2460] follows:

1. IPv6 Header
2. Hop-by-Hop Options Header
3. Destination Options Header
4. Routing Header
5. Fragment Header
6. Authentication Header
7. Encapsulating Security Payload Header
8. Destination Options Header
9. Upper-layer header

The headers can be chained and used multiple times almost without restrictions. [*RFC* 2460] states that "IPv6 nodes must accept and attempt to process extension headers in any order and occurring any number of times in the same packet, except for the Hop-by-Hop Options Header which is restricted to appear immediately after an IPv6 header only." Sending bogus or endless combinations may lead to increased resources consumption and eventually to DoS. The headers may be also crafted in a way to bypass security systems. Additionally, there are some specific threats linked to Hop-by-Hop Options Header and Routing Header.

Hop-by-Hop Options Header First extension header to appear in a packet, if present. It contains information that must be processed on every intermediate node. Structure of the header is outlined in Figure 4.2.

One of the options is the Router Alert option which indicates that a router should inspect the packet as the information is carries may be valuable for the router. Flood of packet with this option on may decrease performance or even cause denial of service. According to [21], the Router Alert option can be misused to bypass access list (ACL) restrictions. [21] describes a situation, where ICMPv6 Echo Request message with Router Alert option bypassed ACL restriction. The request was let through although it was forbidden by ACL.

Another possibly problematic option is the Jumbo Payload option. IPv6 jumbograms defined in [*RFC* 2675] are packets that carry payload bigger than 65 535 octets (up to one byte less than 4GiB). These packet can be misused to cause DoS by consumption of resources. Proper inspection of jumbograms will very likely be challenging for security systems such as firewalls and IDS/IPS. Furthermore, most of the IPsec implementations do not support jumbograms [5].

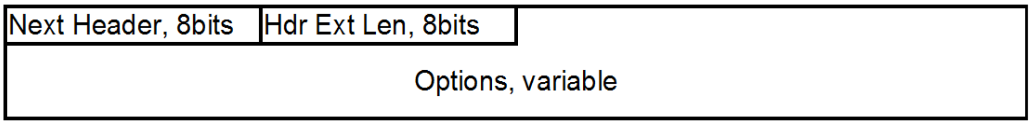


Figure 4.2: Format of Hop-by-Hop Header

Routing Header This type of header, with structure as outlined in Figure 4.3, is used to list intermadiate nodes the packet should pass through on its way to the destination. Currently, there are three types of Routing Header - Type 0, 1 and 2. Type 2 is used fro Mobile IPv6, Type 1 is used by a DARPA project and Type 0 is currently deprecated by [*RFC* 5095] due to severe security ramifications. It could be used to launch MITM or DoS attacks, bounce traffic off a host to bypass security restrictions, etc. Packets with Type 0 Routing Header (RH0) must not be proccessed by nodes and is no longer required to be implemented.

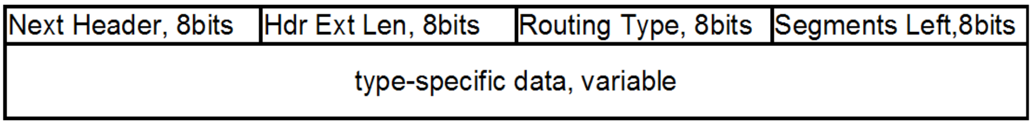


Figure 4.3: Format of Routing Header

4.2.2 Neighbor Discovery

As it has been already mentioned in this work, *Neighbor Discovery Protocol* (NDP) is a replacement of Address Resolution Protocol (ARP) which is based on elements of ICMPv6. ICMPv6 is inseparable part of IPv6 protocol and cannot be completely filtered out. Recommendation for filtering can be found in Attachment II. Elements of NDP provide for:

- autoconfiguration, prefixes and other configuration,
- Duplicate Address Detection,
- ARP-like address resolution,
- neighboring routers discovery,
- Neighbor Unreachability Detection,
- and redirection.

Most of these mechanisms can be exploited for malicious activity. Formerly, no additional security to this mechanism was introduced. The NDP itself is vulnerable to different types of spoofing, redirection, replay and DoS attacks. The attacks will be descibred in detail in Chapter 6.3. IETF later on specified *Secure Neighbor Discovery* (SEND) in [*RFC* 3971]. SEND uses *Cryptographically Generated Addresses* (CGA) defined in [*RFC* 3972] to improve security of NDP. Please refer to Figure 4.4 for a schematic outline.

CGA are based on asymetric cryptography, namely RSA algorithm. When using CGA, the lower 64 bits of their IPv6 address are generated from the network prefix, random number and public key using SHA-1. The parameters are then sent by NDP so it can be verified by communication partner. The whole SEND message is digitally signed. However, CGA may also be exploited for DoS attacks. Please refer to Section 6.3.7.

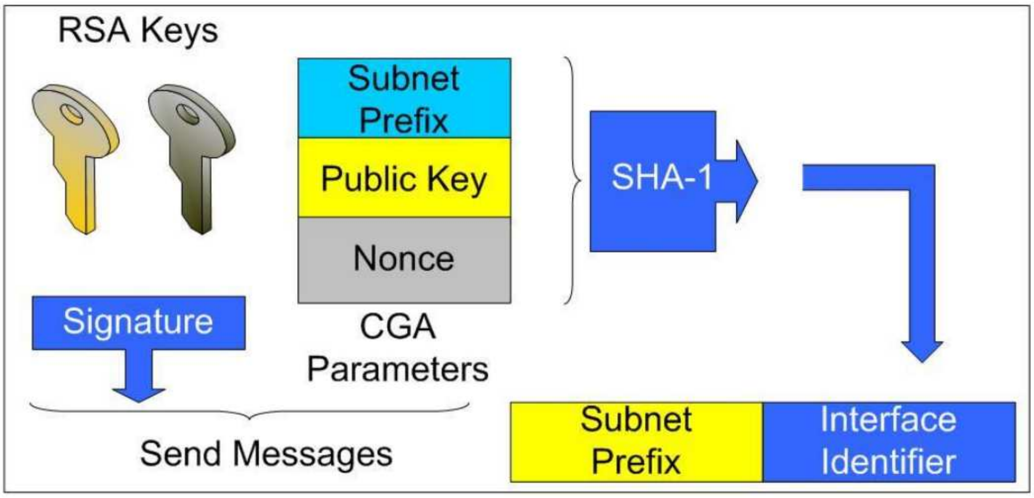


Figure 4.4: Cryptographically Generated Addresses [5]

4.2.3 Quality of Services

The threats associated with IPv6 QoS explicit improvements are not of high severity. The header fields *Type of Service* and *Flow Label* are not protected from tampering, although [*RFC* 3697] specifies them as non-alterable. Adversary could gain benefits by modifying these fields while in transmit, leading into fraudulent use of prefered traffic streams. Firewalls, ACLs and IDS/IPS solutions should not make decisions based only on these fields. QoS can be used together with IPsec so it should be taken into account that information about upper layer protocols may not be accessible for inspection.

4.2.4 Mobile IPv6

Although MIPv6 was designed with security as a primary concern [5], several opportunities for malicious activity were left open. MIPv6 is susceptible to wide range of attacks such as rogue home agent, man-in-the-middle threats, interception, hijacking, spoofing and DoS attacks. As the nodes are moving, centralized security systems are bypassed so security of the mobile devices should be put into the spotlight.

Most attacks involve modification or forging *Binding Update* (BU) messages, IPv6 headers, home or *Care-of Address* (CoA) [5]. Attacks including DoS opportunities, taken from [5], are as follows:

- Inducing extra BUs with bogus CNs (*Correspondent Node*). Although no satisfactory defense exists, route optimization is optional, and the tradeoff is to risk suboptimal routing. A MN (*Mobile Node*) can be selective about route optimization.
- Preventing a legitimate BU from completing while sending bogus BU to CN (where the attacker is on the same link as the victim).
- Reflection attacks, whereby the victim's address is forged as the source, so that the victim is flooded with replies.
- Replaying old route optimization BUs, especially if sequence numbers are unreliable because of crashes or rollover.
- Bypassing firewall egress filtering with a forged Home Address Option.

4.2.5 IPv6 Latent Threats in IPv4 Networks

Last in this section are IPv6 latent threats in IPv4 only networks. These threats should be mentioned in this work as well, becease they were introduced together with IPv6 support on network devices and operating systems. The fact, that IPv6 is not in use on particular network does not mean it should be ignored. As long as network devices understand the protocol and the features are not turned off, an attack can be performed over IPv6.

The following actions can be performed by an IPv6 capable node on an IPv4 network. List is taken from [2].

- **Roam to an IPv6-enabled wireless hotspot:** The *Router Advertisements* (RA) sent by the wireless router immediately connect the host to the IPv6 Internet.
- **Receive a forged RA messages:** The host is configured to use IPv6 (albeit with only local connectivity if the attacker does not forward the IPv6 traffic to the Internet).

- **Use a routable IPv4 address:** Enables 6to4 connectivity to the Internet (assuming that there is no firewall blocking protocol 41).
- **Existence of the DNS name of isatap.example.org:** Initiates an ISATAP tunnel to this name (again assuming that there is no firewall blocking protocol 41).
- **Teredo tunnel to connect to an IPv6-only mode:** If the NAT/firewall devices allow outbound UDP packets and if the NAT function is quite open (not applicable to IOS routers), a Teredo hole is punched in the firewall and allows every IPv6 Internet machine to connect to the Teredo client.

Once connected to IPv6 network, all IPv6 security considerations applies to the node. It can face any of the threats mentioned in this chapter as well as dual-stack or transitions mechanisms (tunnels) related threats. These are, however, out of the scope of this work.

4.3 Indifferent Threats

Attacks which were not significantly altered by IPv6 introduction are listed and briefly discussed in this section.

Application and Other Layers Attacks This paragraph covers all attacks outside the third layer of the ISO/OSI model. These layers remain untouched by IPv6 adoption, so the same considerations applicable for IPv4 networks are applicable in IPv6 environment as well.

Flooding During this attack, a network defice is flooded with more traffic that it is able to process. This leads to a *denial of service* (DoS) situation. Any IPv6 network faces the same challenges in the matter of defence againts flooding attacks as an IPv4 network.

Man-in-the-middle A *man-in-the-middle* (MITM) attack is act of eavesdropping on the network communications. It is often part of the gaining access phase of an attack. The mechanism is outlined in Figure 4.5: an adversary positions themself in the middle of communication stream (2), while the originally communicating entities still believe they communicate directly (1). The data can be modified or misused. Countermeasures for IPv6, such as IPsec or strong data encryption and mutual authentication, are the same as for IPv4. IPv6 functionalities may introduce new means to accomplish MITM attack.

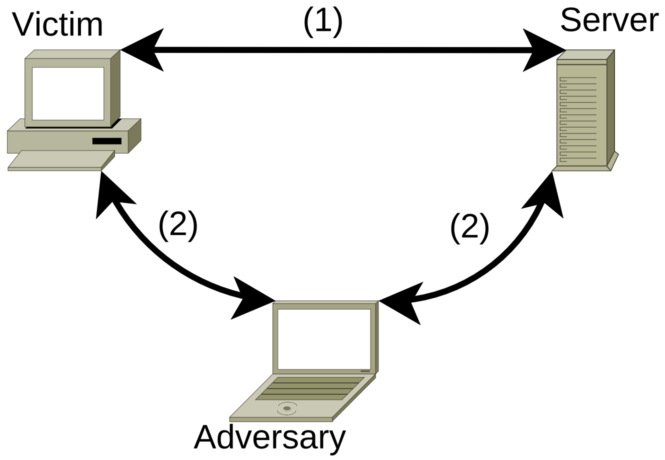


Figure 4.5: Man-in-the-middle Attack Scheme

Rogue Device Any unauthorized device on the network is called a *rogue device*. The most common rogue device threat, often called *evil twin*, is unauthorized wireless access point (WAP) placed on local area network (LAN). This type of threat is not changed for IPv6.

Sniffing A *sniffing attack* occurs when an adversary tries and succeeds to capture network traffic without authorization. The captured traffic can be then used for data analysis or replay attack. Alike IPv4, the only mechanism to protect data transported over the network in IPv6 is encryption.

Distributed Denial of Service This kind of attack is very similar to abovementioned flooding attack, which is often refered to as denial of service attack. *Distributed denial of service* (DDoS) attack leads to bandwith or resources exhaustion as well but involves more then one attacking machine (usually hundreds to thousands). These machines were infected by malicious software which makes them "listen" to adversary's commands. When an adversary orders, the whole group of machines, called *botnet*, starts flooding the target.

This type of attack remains present in the IPv6 world. Furthermore, IPv6 addressing makes it possible for more devices to join the Internet which can result in even more powerful DDoS attacks.

Fragmentation Threats Fragmentation in IPv4 was often used to bypass security systems and to hide attack patterns. Fragmentation as we know it does not exist in IPv6 where fragmentation by intermediary nodes is prohibited [*RFC* 2460]. However, packets may be fragmented by the source node and therefore adversary can use the same techniques to obfuscate attacks. Only minimum MTU differs, for IPv6 it is 1280 octets, and every fragment has to contain a *Fragment Header* (outlined in Figure 4.6). Packets smaller than minimum MTU should be dropped unless it is the last fragment (*More Bit*, represented as `M` in Figure 4.6., is set to "`0`" value for the last fragment).

Next Header,8bits	Reserved, 8bits	Fragment Offset, 13bits	Res,2bits	M,1bit
Identification, 32bits				

Figure 4.6: Format of Fragment Header

5 IDS/IPS Technology Overview

Intrusion detection and/or prevention system, IDS/IPS [1], is a transparent, complementary security solution (hardware or software) to firewalls. It is an important security capabilty designed to detect (and potentionaly react to) the presence of unwanted activity in real-time.

The difference between IDS and IPS is, in most cases, in the settings of particular appliance. However, IDS and IPS can be two different solutions. It will be refered to IDS/IPS as to a one appliance hereafter. Generally, IDS are used in a passive way to only detect potential problems. Typical IDS responses could be increased logging granularity or administrator notification. IPS are used in more active way, to detect unwanted activities and prevent adversary attempts from becoming successful. They can interact and interfere with communications considered as unwanted. Both types also differ in the way they are deployed (refer to Figure 5.1). IDS are often placed outside the main communication stream, f.e. on mirrored ports. IPS must be placed in-line (all incoming data streams have to through the appliance) in order to be able to interfere with the transmissions.

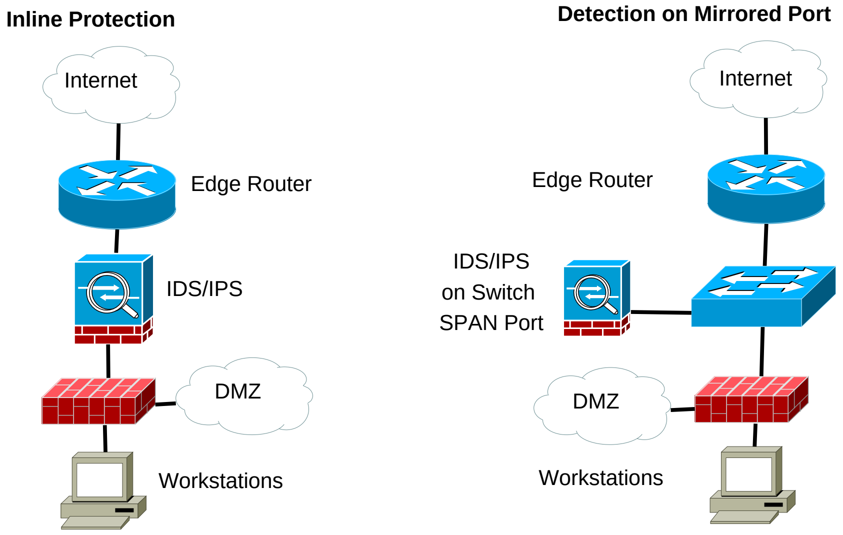


Figure 5.1: Possible Deployment Schemes

[1] By IDS/IPS in this work is meant Network Intrusion Detecion System/Network Intrusion Prevention system, often refered to as NIDS/NIPS. There is also host-based IDS/IPS, sometimes refered to as HIDS/HIPS.

The appliance can be placed both in front of and behind a firewall. The concern here is if we want to detect and/or prevent external or internal threats towards DMZ. Ideal but costly solution is to obtain two appliances and place one in each direction.

IDS/IPS solutions are most reliable for detecting attacks at upper layers of ISO/OSI model or network-focused attacks, such as *denial of service* (DoS) attacks where adversary tries to exhaust bandwith capacity; but it can detect bug exploits, flaw and port scanning. It monitors traffic patterns, scans header informations and examines the contents of packets to detect any malicious content and impending security breaches or attacks. Based on how it is configured, it can react in real time and take action to reduce potential damage. Most common responses are to drop packets, disconnect sessions, reset connections, shutdown the server, trigger alerts and so on.

Additionally, IDS/IPS appliances are can be used for auditing purposes. In other words, they just detect if particular software or protocol is in use on observed network. It then logs source and destination addresses, number and time of occurences and even the certain packets.

There are three commonly used detection mechanisms available: *behavior-based*, *signature-based* and *anomaly-based.*

Behavior-based Is a mechanism, which watches the ongoing network activity and looks for suspicious events. In other words, behavior-based detection is baselined on everyday activity and looks for anything that deviates. This technology allows to detect any differece, including uknown issues such as zero-day attacks. Baseline establisment is considered to be rather difficult process, to define what is "normal" can be challenging.

Signature-based This detection mechanism compares event patterns against known attack patterns, *signatures*, stored in the appliance database. Consequently, its detection capabilty is limited only to known signatures and malicious activity. Therefore, new and zero day attacks can not be revealed. Similarity to antivirus software solutions comes to mind. Regular updates are crucial.

Anomaly-based Detection method, which relies on definitions of all kinds of valid activity is called *anomaly-based.* It is commonly used for protocols, because all the valid forms of a protocol are known and clearly defined in RFCs. Deviations from those forms are then identified as anomalies. Drawback of this method is obvious - just because the traffic follows defined standards, the content can not be considered as not malicious.

5.1 IPv6 Considerations

For the IDS/IPS sensors, IPv6 is just another protocol they have to "understand"; vendors have to provide new signatures. Sensors need to be powerful enough to process all the IPv6 headers, which can possibly be chained. Parsing through all the combinations of extension headers and accurate interpretation of options in these headers will probably increase the CPU demands.

The extended range of IPv6 address space can bring a new challenge once again. Defining rules for unwanted traffic using so-called black-listing can be extremely demanding on accuracy and system resource as well. Shift towards white-listing could be expected.

IPsec, the main security feature of IPv6, has two sides. Native encryption support for end-to-end communication hides its content not only from the adversary but from any kind of packet inspection efforts as well. Although a hardware cryptographic acceleration can be utilized for decrypting packets on-the-fly at high transfer rates, sharing keys of every host on the network with the IDS/IPS is unrealistic. It would create single point of interest for adversary and therefore expose the network to a unnecessary security risk. Because of this, it could be expected that future IDS/IPS solutions would focus more on flow monitoring and analysis. As mentioned in [3], if there is lively data exchange with porn site from a company's computer, the particular content is not of a high concern.

5.2 Current State

There are six major vendors of IDS/IPS appliances on the market according to SANS Institute [6]: Cisco Systems, McAfee, Juniper Networks, IBM, Sourcefire and Tipping Point. The report is rather old, dated to November 2009, and changes have occured since then. Tipping Point was acquired by Hewlett-Packard and Check Point Software Technologies should be considered too. What do they claim about IPv6 support?

Check Point Software Technologies IDS/IPS from Check Point utilize "Hybrid Detection Engine which provides full IPv6 support, ensuring that all attacks currently obfuscated by channeling through IPv6 will be prevented" [13].

Cisco Systems Networking icon, Cisco, claims to have "more than 10 years of IPv6 experience" [14] and that their "organization leads the way by developing standards on the Internet Engineering Task Force (IETF) and integrating IPv6 into its services and technology portfolio" [14]. First traces of IPv6 support can be found in release notes of IPS sensors software in 2009, probably only because older release notes are not available on the company's website.

Hewlett-Packard The HP's flagship, HP N Platform Next-Generation Intrusion Prevention System supports a broad set of traffic types. "It provides uncompromising IPv4 and IPv6 simultaneous payload inspection and support for related tunneling variants (4in6, 6in4, 6in6)" [8]. IPv6 support has been metioned in product sheets for all the New Generation appliances since 2011.

IBM The company claims IPv6 support for IBM Security Network Intrusion Prevention System (formerly IBM Proventia) G series since 2010 when firmware release 4.1 was introduced (current release is 4.6). IPv4 is in the materials mentioned as "legacy". From the [7]: "With release 4.1 of the IBM Proventia Network IPS firmware, attacks will be identified and mitigated for IPv6 network traffic as well as legacy IPv4 traffic".

Juniper Networks Juniper's IDP series intrusion detection and prevention appliances run the latest OS 5.1. Excerpt from the software documentation speaks for itself [9]: "IDP Series devices do not support inspection of IPv6". IPv6 packets are dropped by default, passed through only in so-called bypass mode.

McAfee McAfee's all network security platforms from M-series support IPv6 as of 2013 materials [10].

Sourcefire Next-Generation IPS appliances from Sourcefire support IPv6. In [11] they claim that "Since 2003, Sourcefire has been aggregating network intelligence to provide "context" to network security defenses", where IPv6 is also mentioned. Sourcefire's IPS use open source Snort detection engine, which is said to support IPv6 since version 2.8.0 released in 2009 [12].

These statements are not actually specific. Phrase *"IPv6 supported"* does not say much about the level of reliability and correctness of inspection-based decisions. As these materials come mostly from the marketing department, testing of particular solution is needed before acquisition; either on the customer's side (e.g. as a part of proof of concept) or on the vendor's side (e.g. during presales activities). It will be goal of the subsequent part of this work to define guidelines for such testing. It shall cover the fundamental vulnerabilities of IPv6 which definitely should be recognized by every commercial IDS/IPS solution.

6 IPv6 Attacks and Exploits

This chapter will cover the most effective currently known IPv6 attacks. It will be explained how the attacks work and how to perform them on a network so this knowledge can be based upon during the testing in the last chapter.

The attacks will be divided, quite unconventionally, not according to typology (DoS, MITM, etc.) or location of the adversary (local and remote) but into three plus one certain categories which better serve the purpose of this work. Namely:

- **Reconnaissance** - Section 6.1,
- **Attacks over IPv6** - Section 6.2,
- **Attacks over ICMPv6** - Section 6.3

and one additional chapter which does not discuss particular attacks but implementation imperfections which are important and lively phenomenon of the new protocol, **Implemenatation Maturity Problems** - Section 6.4.

6.1 Reconnaissance

Only network scanning will be discussed in this section as port scanning and other kinds of information gathering have not changed for IPv6.

6.1.1 Network Scanning

It has been stated in the Section 4.1.1 that network scanning is not feasible in IPv6 world. This is only partially true. Network scanning is indeed not feasible when the same techniques used for IPv4 networks are adopted. Simple brute force ping sweeps are not sufficient anymore. In IPv6 environment there are two separate areas of considerations for network scanning, namely *local scanning* and *remote scanning.*

Local scans remain still rather easy. The address space is too vast and there are no broadcasts. Multicast can not be simply pinged as per [*RFC*4443]. This is, however, true for hosts compliant to it. It can be said about only one from the two most widely used operating systems, Linux and Microsoft Windows. Suprisingly, it is Microsoft Windows but they adopted [*RFC*4443] with its exceptions as well and therefore the multicast can be successfully pinged. More will be discussed in Section 6.3.8. Local reconnaissance can be acomplished by other means as well. When an adversary has access to LAN, they can perform passive discovery. It is possible to listen for messages from DAD and ND mechanisms and collect IP addresses. Sometimes a few addresses would be enough to discover the numbering pattern and selectively ping the hosts.

Remote scans became more complicated but several ways to speed up the scanning process exist. IPv6 addresses in the real world deployment are mostly not random [23] [24] [*RFC DRAFT*1]. Numbering conventions often used can be listed as follows.

SLAAC-based The "unknown" part of these addresses is really the lower 64 bits which are based on MAC address of the node. The construction is outlined in Figure 6.1. First 24 bits are unique identifier of the vendor of machine's network interface card. These are known (e.g. for a virtual infrastructure) or guessable using a dictionary of these values. Next 16 bits are constant and the truly unknown bits are the lowest 24 which makes the scanning much faster.

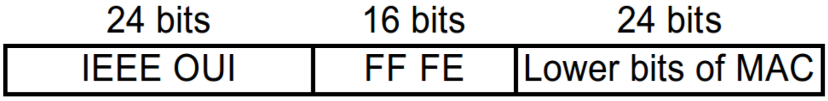


Figure 6.1: Lower 64 bits of SLAAC-based IPv6 Address

IPv4-based These addresses are likely used in dual-stack environment and contain IPv4 address in the IPv6 address. An exmaple could be `2001:db8::192:168:1:1`. This makes the search space same as in case of IPv4 environment.

"low byte(s)" Only the lower byte or two are used for host numbering. The search space in this case is 2^8 or 2^{16}.

“wordy” Wordy addresses such as `2001:db8::b00b:babe` or `2001:db8::dead:beef` are easy to remember but easy to guess as well. Some kind “dictionary” scan can be utilized as well.

“service_port” Addresses used on machines dedicated to a single service often use easy to remember addresses such as `2001:db8::80` for web server. This addressing scheme makes the search space as small as 2^8.

Another kind could be addresses provided by DHCP. Once one host is found it would be easy to discover pattern of the DHCP pool but a speck of luck is needed. When scanning is not suitable, an adversary will very likely focus on DNS servers or particular types of traffic, such as e-mails, from which the addresses can be extracted. If DNS is in use on a network, a kind of dictionary attack may be utilized as there are common naming conventions for servers such as capital cities, gods from greek mythology and so on.

6.2 Attacks Over IPv6

This section discusses attacks that exploit features of IPv6 itself. Embedded ICMPv6 mechanisms are discussed in the next, separate and more comprehensive section.

6.2.1 Extension Headers Exploits

Extension headers seem to introduce a whole new attack domain to IPv6. Not only they may cause concern about the performance of security systems that have to process the headers correctly but security researchers have already found several ways to exploit the extension headers.

Hop-by-Hop Options Header is the one and only extension header which has strictly defined position in the IPv6 packet. It has to be placed right after the IPv6 header and has to be present only once as it is the only extension header that is being inspected on every intermediate node. One of the options that can be defined within Hop-by-Hop header is Router Alert Option which informs routers on the path that they should closely examine the content as it could countain information valuable for them, such as RSVP or MLD message [*RFC* 2711]. The option itself is specific *Type-Length-Value*

(TLV) encoded number within Options field of the header as outlined in Figure 4.2. Unfortunately, this option can be exploited to cause DoS attack on a router. Router spends more time examining content of packets with Router Alert Options. Therefore, the situation when the router is flooded with large number of these packets can lead to inadequate resources consumption or deterioration of response time.

Extension headers are very useful when it comes to firewall evasion. Particular techniques described in [21], [22] and [26] will be discussed in detail in Section 7.4.2 where additional firewall testing approach is described.

Another security solution which can be rather easily evaded using extension headers is *Router Advertisement Guard* (RA Guard) [27]. RA Guard is a solution intended to protect against Router Advertisement attacks which will be discussed in Section 6.3. All traffic on LAN has to pass through RA Guard in order to make the protection effective. RA Guard can be a standalone solution but it usually is additional functionality of switches. Assessing RA Guard is out of the scope of this work.

Extension header threats are closely linked to fragmentation attacks because every fragment employs Fragment Header which is extension header as well. Attacks associated to Fragment Header, respectively fragmentation itself will be discussed in the following section.

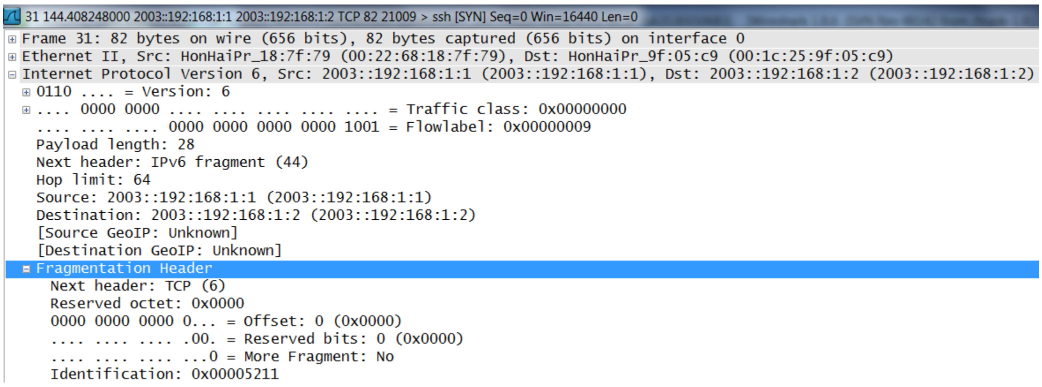
6.2.2 Fragmentation Attacks

Fragmentation attacks are well known from IPv4 already but IPv6 changes the fragmentation philosophy. Fragmentation can be perfomed exclusively by the source host and *not* on intermediary nodes [*RFC* 2460]. This surely benefits the ease of transmission but adversary can craft fragments more accurately. Fragments can be used to bypass IDS/IPS systems as well as firewalls. The techniques for hiding attack patterns or evading security systems are [25]:

- **evasion** - inserting fragment which is not processed by IDS/IPS but let through due to its transparency,
- **insertion** - inserting fragment which is accepted by IDS/IPS but discarded by a target host,

- **overlapping fragments** - overlapping fragments could cause DoS during reassembly or misinterpretation of the data thus hiding attack pattern,
- **tiny fragmentation** - attempt to hide attack pattern; huge amount of tiny fragments is a sign of a coming attack,
- **disordered arrival of fragments** - disordered fragmnents of several packets arriving at once is a technique trying to avoid deep packet inspection,
- **fragment flooding** - another strategy designated to avoid deep packet inspection.

Handling of IPv6 fragments is described in [*RFC* 2460] and updated by [*RFC* 5722]. All overlapping fragments should be silently discarded including these not yet received. However, none of the current versions of mainstream operating systems complies to these RFCs [25].



```
31 144.408248000 2003::192:168:1:1 2003::192:168:1:2 TCP 82 21009 > ssh [SYN] Seq=0 Win=16440 Len=0
⊞ Frame 31: 82 bytes on wire (656 bits), 82 bytes captured (656 bits) on interface 0
⊞ Ethernet II, Src: HonHaiPr_18:7f:79 (00:22:68:18:7f:79), Dst: HonHaiPr_9f:05:c9 (00:1c:25:9f:05:c9)
⊟ Internet Protocol Version 6, Src: 2003::192:168:1:1 (2003::192:168:1:1), Dst: 2003::192:168:1:2 (2003::192:168:1:2)
  ⊞ 0110 .... = Version: 6
  ⊞ .... 0000 0000 .... .... .... .... .... = Traffic class: 0x00000000
    .... .... .... 0000 0000 0000 0000 1001 = Flowlabel: 0x00000009
    Payload length: 28
    Next header: IPv6 fragment (44)
    Hop limit: 64
    Source: 2003::192:168:1:1 (2003::192:168:1:1)
    Destination: 2003::192:168:1:2 (2003::192:168:1:2)
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
  ⊟ Fragmentation Header
      Next header: TCP (6)
      Reserved octet: 0x0000
      0000 0000 0000 0... = Offset: 0 (0x0000)
      .... .... .... .00. = Reserved bits: 0 (0x0000)
      .... .... .... ...0 = More Fragment: No
      Identification: 0x00005211
```

Figure 6.2: IPv6 Atomic Fragment

IPv6 introduces Fragment Header (structure outlined in Figure 4.6) which is used to describe fragments and carries information needed for reassembly. Together with this extension header is introduced so-called Atomic Fragment. It is a packet that contains Fragment Header although it is not fragmented - offset value and more bit is set to all zeroes, please refer to Figure 6.2. Atomic Fragments may be exploited for security systems evasion. Handling of these packet was standartized in May 2013 and overlapping fragments were explicitly forbidden in December 2009. Currently, there are some rummors in the IPv6 security community that IETF is trying to workout mechanism to *completely* remove fragmentation from IPv6 because of its security concerns.

Moreover, incomplete stream of fragments may be exploited for amplification or reflective DoS attacks. When systems are flooded with incomplete fragments stream they wait for specified amount of time for the rest of the stream to arrive. If the fragments do not arrive, host sends back ICMPv6 *Time Exceeded (fragment reassembly time)* message.

6.3 Attacks Over ICMPv6

As it has been already mentioned, ICMPv6 is a vital part of IPv6. It provides for several handy features and replaces ARP utilized in IPv4. Unfortunately, it also seems to be an Achilles' heel of the whole protocol as many attacks target it. Nevertheless, it has been always taken into account that the adverasy needs access to LAN to exploit these ICMPv6 vulnerabilities. In most cases, these threats have to be considered as an insider threats and public LANs as networks of much higher risk.

Following sections of this chapter will discuss selected currently known ICMPv6 vulnerabilities and associated exploits, sorted in random order.

6.3.1 Duplicate Address Detection Attack

Duplicate Address Detection (DAD) is a mechanism employed by hosts and *Stateless Address Autoconfiguration* (SLAAC) feature of IPv6 to prevent duplicate addresses on a network. It is susceptible to *Denial of Service* (DoS) attack.

When host is to join a network with address acquired, for example, through SLAAC, it sends an ICMPv6 *Neighbor Solicitation* (NS) message to all nodes multicast address, `FF02::1`, in order to verify that there is no host already in possession of this address. If there is no reply in specified time, the hosts assumes the address is not in use and starts using it as its own IPv6 address. This mechanism is illustrated in Figure 6.3 by steps *(1)*, *(2)* and *(3)*.

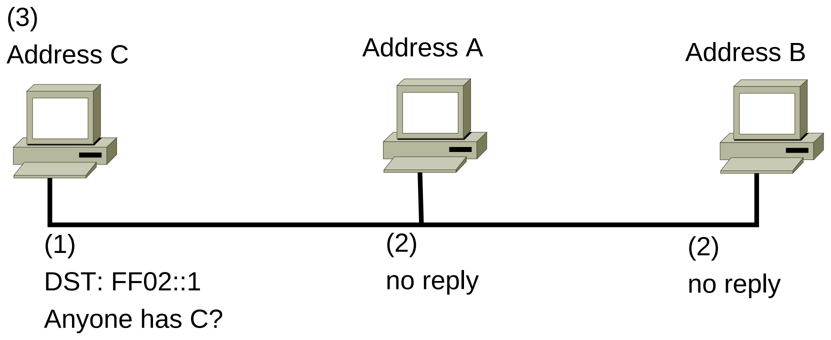


Figure 6.3: Proper Duplicate Address Detection

However, anyone can reply to NS message claiming that the particular address being solicitated is their address. When an adversary has access to LAN, therefore is recipient of all nodes multicast messages, there is nothing that could stop them from interrupting the DAD mechanism with malicious activity. The principle is very simple. Anytime a host wants to join the network and sends NS message to all nodes multicast address, adversary responds claiming the address is theirs thus preventing any new hosts from joining the network. The attack is summarized in Figure 6.4 by steps *(1)*, *(2a)*, *(2b)* and *(3)*.

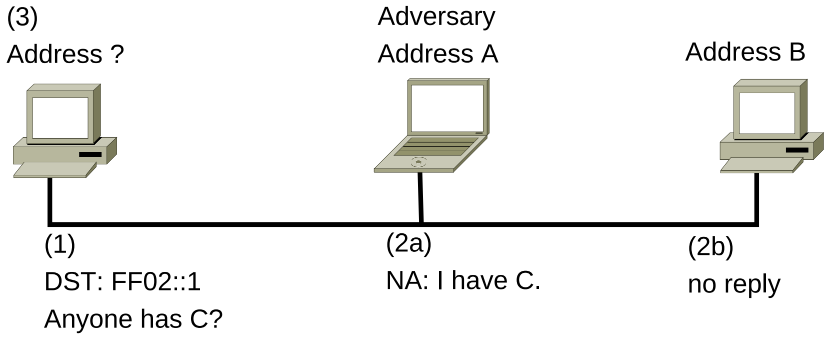
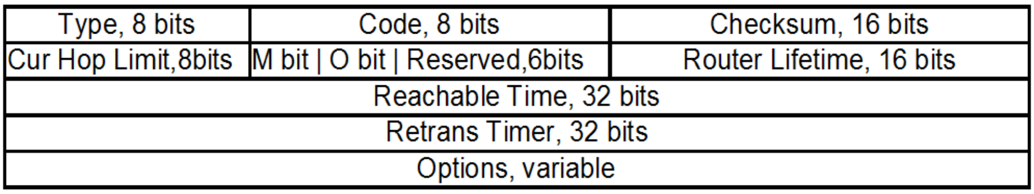


Figure 6.4: Duplicate Address Detection Attack

6.3.2 Router Advertisement Spoofing

Router Advertisement (RA) is a type of ICMPv6 message (Type 134) which is periodically sent by a router in order to advertise itself and a particular subnet settings to all nodes multicast address `FF02::1`. It can also be request from the router by any node on the subnet by sending Router Solicitation message to all routers multicast address `FF02::2` (please refer to Attachment I). Structure of a RA message is outlined in Figure 6.5.



Type, 8 bits	Code, 8 bits	Checksum, 16 bits
Cur Hop Limit,8bits	M bit | O bit | Reserved,6bits	Router Lifetime, 16 bits
Reachable Time, 32 bits		
Retrans Timer, 32 bits		
Options, variable		

Figure 6.5: Format of Router Advertisement Message

It is obvious that RA messages can be arbitrary spoofed and thus adversary can set any IP address as a default router and cause either Dos by advertising bogus address or MITM by advertising their own, advertise various network prefixes, DNS servers and so on. Another way to cause DoS is sending spoofed RA message which advertise the current router but with Router Lifetime (see Figure 6.5) value set to zero. This will force all nodes on the subnet to discard the default router. Lets look into the MITM attack in more detail.

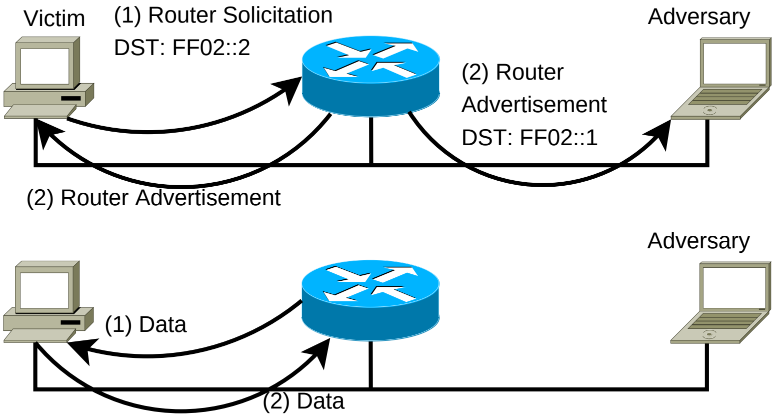


Figure 6.6: Proper Router Solicitiation/Advertisement Mechanism

Proper use of RS and RA messages is outlined in Figure 6.6. Host can request RA by sending RS message to all routers multicast address *(1)*. Router replies with requested RA message *(2)*. Communication then takes place directly between hosts and the router (steps *(1)* and *(2)* in the lower scheme).

However, anyone can send RA message and does not even have to wait for RS request. Both solicitated and periodical advertisements can be overriden by a forged advertisement with higher priority. When host receives the forged RA message, it discards the previously advertised information and replaces it with the one sent by an adversary.

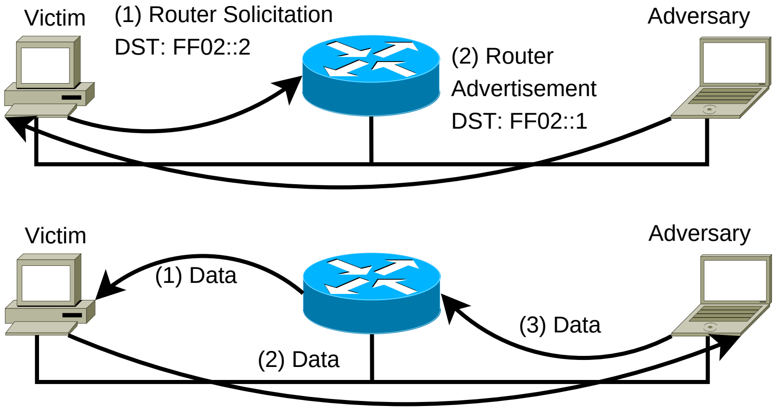


Figure 6.7: Man-in-the-Middle Attack with Spoofed Router Advertisement

The attack is outlined in Figure 6.7. The upper figure depicts attacker answering RS with its own forged RA *(2)*. The setup then opens door for MITM attack for the adversary as all the traffic destined to router arrives directly to them (*(2)* on the lower scheme) as there can be only one default gateway on the network. The adversary can modify or obtain private data, hijack sessions and much more.

6.3.3 Router Advertisement Flooding

Another ICMPv6 flooding attacks is RA flooding (the message is outlined in Figure 6.5). The principle is simple, and adversary floods whole network is just particular host with forged RA messages. The attack can be performed in number of slight modification. The messages can be simple advertisement messages or messages bearing data, such as announcing new route. The victim is overwhelmed by processing the information, resources are exhausted and the situation leads to DoS.

Historically, there were bugs in several operating systems which made this attack more serious. All major operating systems were vulnerable to

RA flooding - Microsoft Windows 2003, 2008, XP, 7 and even Windows 8 were released with the same bug known from around year 2008; Linux, Cisco IOS, Juniper Netscreen, FreeBSD and OS X [21] [22]. The systems crashed, stopped responding or lost its connectivity. When the issue was fixed, just simple modification of the RA message was enough to accomplish the same results [21] [22]. Currently, all the issues should be fixed by the vendors. Microsoft fixed the issue in its operating systems in updates released in April 2013, it is not necessary to reboot the system after RA attack anymore. MS Windows does not respond during the attack as its CPU utiliziation reaches 100% but it recovers when the flooding stops.

6.3.4 Neighbor Advertisement Spoofing

Neighbor Advertisement (NA) ICMPv6 Type 136 messages (with structure as outlined in Figure 6.8) can be spoofed the very same way as RA messages in Section 6.3.2 and the functionality of Neighbor Discovery is not very different from the one of ARP in IPv4. It is obvious that it can be exploited to perform MITM attack on LAN where an adversary can position themself into the data stream between two communicating hosts. The principle of MITM attack of solicitation/advertisement messages has been previously discussed in Section 6.3.2.

<table>
<tr><td>Type, 8 bits</td><td>Code, 8 bits</td><td>Checksum, 16 bits</td></tr>
<tr><td>R bit | S bit | O bit |</td><td colspan="2">Reserved, 29 bits</td></tr>
<tr><td colspan="3">Target Address, 128 bits</td></tr>
<tr><td colspan="3">Options, variable</td></tr>
</table>

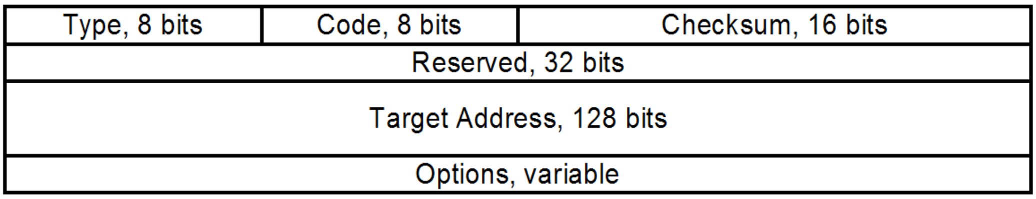
Figure 6.8: Format of Neighbor Advertisement Message

If an adversary intends to perform MITM attack with forged NA messages, they do not have to wait for a solicitation request first. The mechanism distinguishes between two kinds of NA messages, namely solicitated and unsolicitated. These two differ in the value of Solicitated flag (represented as S bit in Figure 6.8). Furthermore, an adversary can set Override flag (represented as O bit in Figure 6.8) that forces the target host to overwrite an existing neighbor entry in its cache.

Less sophisticated than use of forged NA messages for MITM is its use for DoS attack. In this case, an adversary just passively monitors the traffic for NS messages and answers every single one of them claiming to be the host in question. It can be used to gather data or just to discard them and cause DoS as the hosts would send data to the adversary and would not be able to communicate between themselves.

6.3.5 Neighbor Solicitation Flooding

Another DoS attack is a Neighbor Solicitation (NS) flooding attack. NS message is a Type 135 ICMPv6 message and it is properly used to obtain MAC address when only IP address is known. Refer to Figure 6.9 for outline of the message format. Host that needs to perform this kind of address resolution send NS message to all nodes multicast address `FF02::1`. Host which is in posession of said IP address responses with Neighbor Advertisement message containing its MAC address.



Type, 8 bits	Code, 8 bits	Checksum, 16 bits
Reserved, 32 bits		
Target Address, 128 bits		
Options, variable		

Figure 6.9: Format of Neighbor Solicitation Message

When an adversary floods a victim with NS messages, it has to process them all and appropriately response to each and single one of them. Scheme of the attack is in Figure 6.10. The flooding leads to resources exhaustion and therefore to DoS situation. All major operating systems are vulnerable to this type of attack [21] including Microsoft Windows 7. The attacks do not affect the source of the flood.

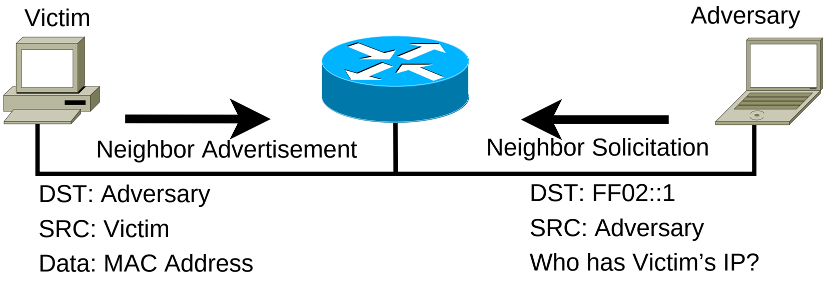


Figure 6.10: Scheme of Neighbor Solicitation Flooding Attack

6.3.6 Link Deterioration

Another DoS attack which, however, is not very effective exploits *ICMPv6 Packet Too Big messages*. Proper use of the messages is to negotiate Maximum Transmission Unit (MTU) of a path. This mechanism is essential for IPv6 because fregmentation on the intermediary nodes is forbidden.

During the attack adversary repeatedly sends illegitimate ICMPv6 Packet Too Big messages to a router and therefore reduces MTU of a link to the minimum of 1280 octets. Consequently, the link is not utilized up to its capacity. The attack is schematically outlined in Figure 6.11.

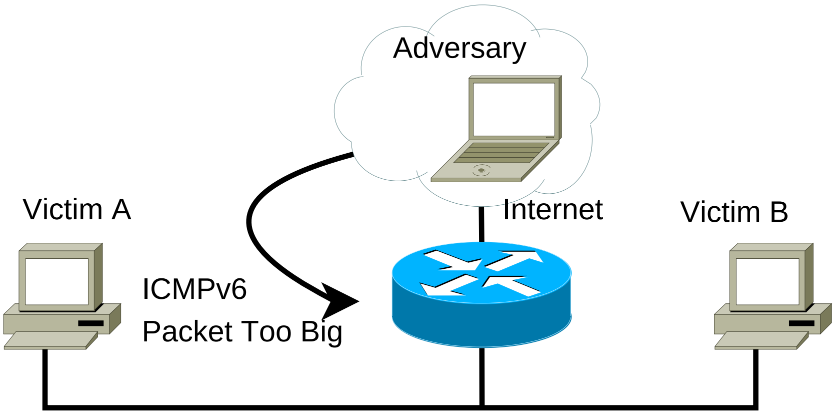


Figure 6.11: Scheme of Link Deterioration Attack

6.3.7 Secure Neighbor Discovery Flooding

Secure Neighbor Discovery (SEND) mechanism is discussed in section 4.2.2. It was developed as a countermeasure to several Neigbor Discovery and Router Advertisement threats, most of which are discussed earlier in this chapter. It introduces new Options which employs CGA signing to authenticate messages. However, no wide deployment of SEND is expected because it requires implementation of Public Key Infrastructure (PKI) which is not trivial and key distribution would cause traffic overhead. Moreover, DoS attack targets SEND. It is a simple flooding attack when adversary sends many, for example, Neighbor Soliciation messages signed with bogus CGA Options. Every host on the network receives the messages as it is sent to all nodes multicast address `FF02::1`. Cryptographic CGA verification steals a lot of CPU time and causes DoS attack. The attack is illustrated in Figure 6.12.

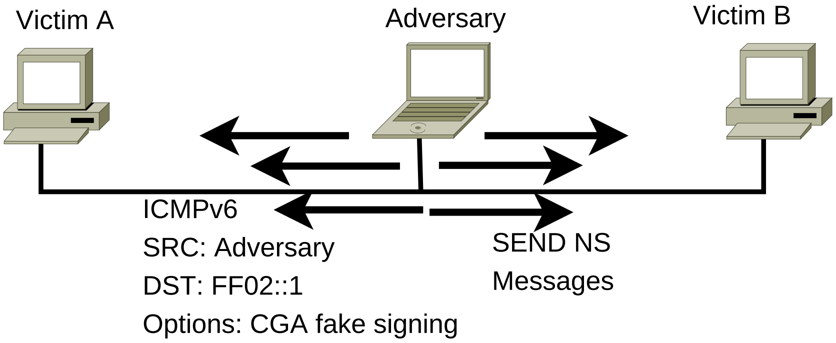


Figure 6.12: Scheme of Secure Neighbor Discovery Flooding

6.3.8 Smurf Attack

Smurf attack which should be called *multicast* amplification attack in IPv6 environment is described in Section 4.1.2. It is mentioned there that the attack should be no issue in environment with nodes compliant to [RFC4443] which forbids responding to ICMPv6 messages with any multicast address as a destination. At the same time, it defines *two exceptions*. One is ICMPv6 Packet Too Big Message (Type 2) which is sent by a router when the packet is larger than MTU of the link leading to destination. Second is any ICMPv6 message with invalid option value which makes every node registered with the multicast address respond with *ICMPv6 Parameter Problem* error reply.

However, the current state is that MS Windows operating systems are compliant to [RFC4443] and attempt to smurf them is degraded to ICMPv6 Echo Request flooding as they do not respond. The flooding leads to increased resources consumption and it can lead to Dos in an extreme case. On the other hand, Linux distributions are known for its non-compliance to [RFC4443] and are vulnerable to smurf attacks and even remote smurf attack can be accomplished due to a flaw in stack implementation [22].

6.4 Implementation Maturity Problems

Before IPv6 functionality is introduced in any security system, developers have to write new kernel (core) from scratch. It is not just about parsing the new protocol structures correctly but also about getting the right context from doing so.

All vendors have come trhough this when developing appliances for IPv4 and many mistakes were made (one example for all - legendary *Ping of Death*). The situation nowadays with IPv6 seems similar to the one of IPv4 in the early days as many bugs and flaws emerge in the IPv6 functionalities of current systems.

One way to measure the maturity of IPv6 implementations could be through *Common Vulnerabilites and Exposures* database.

6.4.1 Common Vulnerabilities and Exposures

Common Vulnerabilities and Exposures (CVE) database maintains list of all kinds of discovered security issues within commonly used products, both hardware and software. Every commonly known information security issue is assigned an unique identifier. Approximately twice as many IPv6 than IPv4 issues is registered every year [21]. This fact may be surprising if one compares the extent of current IPv4 and IPv6 deployment. Currently, there is 142 registered IPv6 associated vulnerabilities in the database of which 7 were registered in 2013 (state as of 1^{st} June 2013) and certainly, there is no guarantee that all issues were registered. Just to mention a few of the vendors or products on the list, in random order: Cisco IOS, Microsoft, Mozilla Firefox, Juniper, IBM, Sun Solaris, Kaspersky Labs and Linux Kernel.

These issues are generally found when the product is placed in real-world production environment. It is obviously the most effective testing approach but definitely not the right one. *Fuzzing* is a technique known mainly from web application testing. It is an effort to find a bug in the application by generating variety of input values, mostly just bogus. Adversary then exploits the flaw for malicious purposes such as injection attacks or just crashing down the application. There are several tools available for IPv6 and ICMPv6 packets fuzzing and even more sophisticated implementation checkers.

7 IDS/IPS Security Assessment

The reasons why IPv6 testing of the IPv6 capable solutions is necessary is explained in Section 5.2. Basic guidelines for testing of IDS/IPS solutions are set up in this chapter. One physical appliance (unlike original intentions, see Section 7.5) is then tested in compliance to these guidelines and the results evaluated therein. It builds on the attacks description from previous chapter.

7.1 Appliance Selection

Two types of IDS/IPS appliance were available for testing. Both will be used in the experimental simulations in this chapter. However, the testing shall be applicable to appliances from any other vendor as well.

First is a software appliance which is designed to be employed in a virtual infrastructure on VMware platform. *Virtual IPS* is characterised by minimal requirements and technical specifications as described in Table 7.1.

Processor	2x Quad Core @ 2.83GHz
Memory	1 GB RAM
Disk Space	10 GB HDD
Operating System	VMware ESX Infrastructure 3 version 3.5
Throughput	Up to 700 Mbps
Segment(s)	1
Connections per second	19 000
Max. number of connections	500 000

Table 7.1: Virtual IPS Specification

Second is a hardware based, stand-alone appliance *Network IPS*. Its technical and performance specification is summarized in Table 7.2.

Both of these appliances utilize current firmware, namely version 0.0, and have been updated with latest signature update package. Every update package consists of currently released signatures which should be always installed upon release. Therefore, both appliances possess the same IDS/IPS capabilities. It is possible to manage these appliances through centralized

Inspected Throughput	800 Mbps
Interfaces (Segments)	4 (2)
Connections per second	35 000
Max. number of connections	1 300 000
Latency	$< 150\mu s$

Table 7.2: Network IPS Specification

management system, a solution primarily intended for centralized administration of multiple appliances. This is, however, optional and both appliances can be managed locally through management console port.

The management console on the physical appliance is accessible via Ethernet cable with RJ45 connector. The appliance has assigned address from `169.254.0.0/16` subnet as per [*RFC* 3927] and the Java-based web interface can be accessed via a web browser. Supported operating systems (OS) are

- Microsoft Windows XP,
- Microsoft Windows Vista,
- Microsoft Windows 7

with one of the following browsers

- Microsoft Internet Explorer 8,
- Microsoft Internet Explorer 9,
- Mozilla Firefox 13

and with *Java Runtime Environment* (JRE) Sun 1.6.x or IBM 1.6.x.

For the virtual appliance, it is possible to access the web interface via the same browsers and OS as mentioned above. The only difference is that the link is virtual and has to be established within the virtual infrastucture.

The experiments will zero in on the *signature-based* detection mechanism. The signatures in these appliances should cover *anomalies* from RFC-defined standards as well and there is no IPv6 network available for the purpose of this work where it could be possible to establish a reasonable baseline for *behavior-based* detection.

7.2 Testing Tools

Very popular linux distribution which by default contains several penetration testing tools used both by professional and enthusiast security personnel is *Kali Linux* (formerly known as Backtrack). Kali Linux is a Debian-based distribution developed by Offensive Security which is claimed to be "the most advanced penetration testing distribution, ever" [20]. It is used as a base for the purpose of this thesis, particularly version 1.0.1.

There are tons of useful tools integrated directly in the Kali Linux. However, only one toolkit focuses solely on assesing IPv6. *THC-IPv6 Attack Toolkit* is very comprehensive suite consisting of tens of adaptable tools. Several implementation bugs and vulnerabilites have been discovered with help of this toolkit; lets mention current vulnerability in handling extension headers which causes crash of Kaspersky Lab's personal firewall [17], similar crash of Aventail Personal Firewall 2012 [21] or ways to bypass security in some Cisco or Zyxel products [21]. The source code comes with easy-to-use library written in C programming language. When being compiled from the source code, library `libssl-dev` is required to be present on the system.

Very similar but not as comprehensive toolkit as the one from THC is being developed by *SI6 Networks*. It consists of rather smaller number of tools where most of the functionalities are the same but it posses extended reconnaissance features. Nevertheless, the toolkit is not integrated into the Kali Linux and may be obtained from the project's Git repository [18]. Additional package `libcap0.8-dev` has to be installed prior compilation.

Where better granularity for packet crafting is needed, tool *Scapy* comes in handy. Scapy is a packet manipulation tool, which can be used to craft specific packets, send or sniff traffic. Whole tools can be based on Scapy, which is interactive python application. Package `python-gnuplot` should be installed for smooth run of the application. Additionally, package `python-sphinx` may be needed in order to generate documentation.

Not only security of the IPv6 protocol itself should be focused on during the testing. As it has been already discussed in this work, immaturity of IPv6 implementations can cause a security threats as well. Both of the above mentioned toolkits contain fuzzers which generate all possible valid and invalid combinations of options in packets. For higher level of assurance, additional tool for assesment of IPv6 stacks will be used in this work. ISIC, *IP Stack Integrity Checker*, contains tools designated for IPv6 as well as IPv4.

It can be obtained from [19] and needs package `libnet1-dev` to be installed prior compilation. Utlizing ISIC, researchers have found bugs for example in Checkpoint Firewall [19].

Fully updated (as of the submission date of this work) Kali Linux VirtualBox image (`.vdi`) with all the above-mentioned tools and their source codes is attached to this thesis. It contains prescripted easy-to-use test cases from the following chapter as well.

7.3 Test Cases

This section will list and describe selected attacks and techniques which will then be used for the testing purposes. It also describes the evaluation approach and scale. The appliance is tested as a *grey box*, with access to settings and management console but without knowledge of the content of particular signatures and details about packets processing. The testing setup is outlined in Figure 7.1.

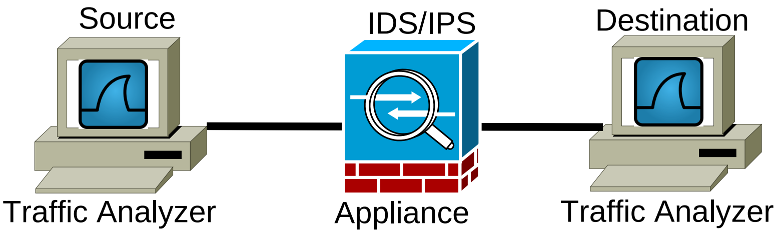


Figure 7.1: Testing Setup Scheme

Every attack scenario, from now on called a *test case* (TC), is assigned a code and a short, characteristic description which will be used in results summary. Following is more detailed description, which is based on the previous *Chapter 6 - IPv6 Attacks and Exploits.*

Description of the IDS/IPS test cases follows. The list does not cover all possible attacks but should suffice for the character of the appliance and purpose of this work. Other findigs, if any, will be discussed in Section 7.5.1.

SCAN-LOC - Local Hosts Scaninng Local network scan exploiting the exception in [*RFC* 4443] as described in 6.1.1 and 6.3.8.

SCAN-REM - Remote Hosts Scanning Remote network scan utilizing classical ping sweeps and speeding up tehniques as described in 6.1.1.

FL1-RA - Router Advertisement Flooding RA flooding as described in Section 6.3.3. Two types of the attack will be examined, first flooding simple RA messages and second flooding RA messages with random route entries.

FL2-NS - Neighbor Solicitation Flooding Flooding with NS messages as described in Section 6.3.5. Again two types of the attack will be examined, first targeting whole network and second targeting just particular host.

FL3-CGA - Secure Neighbor Discovery Flooding SEND attack exactly as described in Section 6.3.7.

RH0 - Handling of Routing Header Type 0 This test case will assess handling of packets with RH0. Packets with this kind of extension header are depreciated as per [*RFC* 5095].

FRAG - Fragmentation Handling The fragments processing should be examined as much as possible in this test case. If not known, it should be reversly engineered to fully understand how the IDS/IPS appliance handles fragmented traffic.

CVE - Common Vulnerabilities and Exposures It is necessary to test as much CVEs as possible. There is currently tool in the THC-IPv6 Attack toolkit tool `exploit6` covering four of them.

FUZZ-IPv6 - IPv6 Fuzzing Fuzzing to assess the correctness and robustness of IPv6 stack implementation. Possible flaws would be very likely found during this test.

FUZZ-ICMPv6 - ICMPv6 Fuzzing Fuzzing to assess the correctness and robustness of ICMPv6 implementation. Possible flaws would be very likely found during this test.

The results will be evaluated according to the following scale in order to maintain uniformity and to provide baseline for benchmarking in case of reuse of these guidelines on another solution.

Passed. Appliance reacted appropriately without significantly increased resources consumption. Expected actions have been taken where aplicable.

Passed, but Findings Noted. Overall performace of the appliance has been satisfactory, however, exceptions were observed; including, but certainly not limited to inadequate resources consumption, temporary instability or inaccurate processing.

Failed. Appliance crashed, did not detect any event or behaved in unexpected and/or unreasonable manner.

For the latter two, detailed explanation and description of symptoms is necessary. It will be attached to the resulting assessment summary. Explanation for "Passed." result is optional.

7.4 Additional Testing

According to system-specific features or deployment-specific requirements, additional testing should be considered. The more is being tested, the higher level of assurance can be achieved.

In this case, both appliances Network IPS and Virtual IPS with firmware 0. has integrated functionality of Data Loss Prevention (DLP) inspection and a firewall. Both functionalities will be assessed.

However, another reason behind this additional testing within this work is that originally intended performance comparison of the physical and virtual appliance was not possible because of the reason mentioned later in Section 7.5 - Test Results.

7.4.1 Data Loss Prevention

DLP is a very broad concept which is completely out of the scope of this work and the additional DLP functionality of the examined appliance is definitely not a complete solution. In a nutshell, goal of DLP is to prevent sensitive data leakage. From its perspective, data belong into three categories:

- data in use - data being used by users, data in production environment;
- data in rest - archived data;

- data in motion - data on a network.

It is clear that data in motion will be the focus. The goal is to prevent data which conform to specific patterns from leaveing company's logical boundaries. The appliance is able to inspect data on the higher layers of ISO/OSI model which has been proofed in practice. This work will challenge Layer 3 inspection by utilizing known covert channels in the IPv6 protocol.

Almost every field in IPv6 packet can be used to hide data [22]. The first one to come up to one's mind is Options field in some of the extension headers. It can be used to smuggle data from the company's premises over the Internet to desired destination. When sent in plaintext, it could be detected by DLP systems but becomes undetectable by pattern matching with added encryption. Figure 7.2 illustrates the idea behind the misconduct.

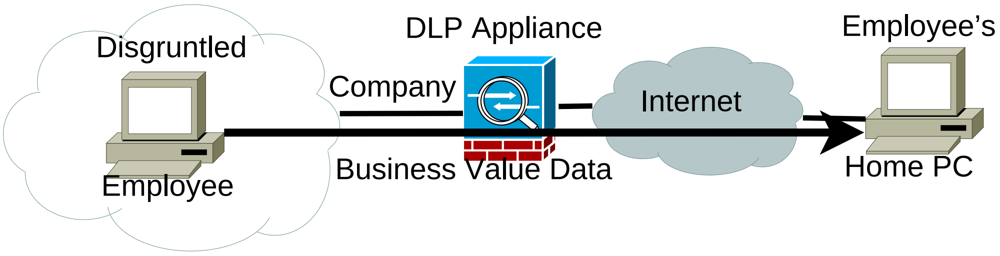


Figure 7.2: Smuggling Data Via IPv6 Covert Channels

7.4.2 Firewall

The integrated firewall supports IPv6 rules and current IPv6 firewalls are, unfortunately, well know for several flaws which provide channels for appropriate evasion techniques. Tool `firewall6` from The Hacker's Choice IPv6 toolkit is the most sophisticated, broadly used tool to perform firewall testing. The tools sends many different types of `SYN` packets.

For the purpose of this work, all traffic destined to SSH port 22 on both TCP and UDP via IPv6 *only* was being blocked by the integrated firewall. Wireshark was run on the target as well as source hosts so detailed analysis of the traffic could be performed.

Individual test cases can be described as follows.

FW1 - Plain Sending Sends a plain, basic `SYN` packet to mak sure the port is blocked.

FW2 - Plain Sending with Data Same as `FW1` but with additional data.

FW3 - IPv4 Ethernet Type Type field at Layer 2 is erroneously set to IPv4 value. This could bypass the IPv6 rule via IPv4 stack if decisions are made based *only* on Layer 2 information.

FW4 - Hop-by-Hop Hdr, Ignore Option Hop-by-Hop Options Header is added to the `SYN` packet but has total length of 8 bytes, Options fields is all zeroes. This packet should be processed the same way as a packet without any extension header.

FW5 - Destination Hdr, Ignore Option In this case, Destination Options Header is used the same way as in `FW4`. The header should have no effect on processing. Format of Destination Options Header is illustrated in Figure 7.3.

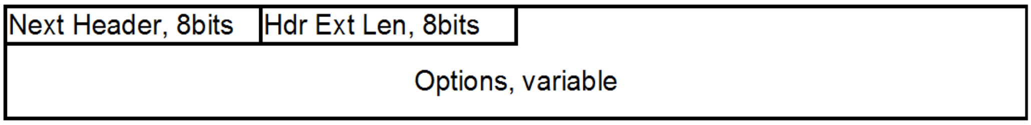


Figure 7.3: Format of Destination Options Header

FW6 - Hop-by-Hop Hdr, Router Alert Packet contains Hop-by-Hop Options Header with Router Alert Option. The option is described in detail in Section 6.2.1.

FW7 - 3x Destination Hdr, Ignore Option This packet contains three Destination Options Headers and the impact should be as mentioned for test case `FW5`.

FW8 - 130x Destination Hdr, Ignore Option Same as `FW7`, resp. `FW5` but the number of headers increases to 130. Some devices may be able to process only limited number of extension headers.

FW9 - Atomic Fragment Packet which contains Fragment Header although it is not fragmented is considered as atomic fragment. IETF standard regarding handling of atomic fragments has been adopted very recently, in May 2013 [*RFC* 6946].

FW10 - 2x Atomic Fragment, Same ID
FW11 - 2x Atomic Fragment, Different ID
FW12 - 3x Atomic Fragment, Same ID
FW13 - 3x Atomic Fragment, Different ID
FW14 - 130x Atomic Fragment, Same ID
FW15 - 130x Atomic Fragment, Different ID
FW16 - 260x Atomic Fragment, Same ID
FW17 - 260x Atomic Fragment, Different ID

All these test cases (`FW11 - FW17`) are variation of `FW9`. They differ in number of Fragment Headers and the Fragment Headers within one packet have either same or different Identification value which is 32-bit within the header as illustrated in Figure 4.6. It is a packet identification value needed for the packet reassembly.

FW18 - 2KB Destination Hdr The packet contains Destination Options Header which has so many options so it has to be fragmented. This could cause firewall to crash.

FW19 - 2KB Destination Hdr+Destination Hdr Extension of test case `FW18`. One regular Destination Options Header is added.

FW20 - 32x 2KB Destination Hdr Another extension of test case `FW18`. The number of headers extends to 32 and it takes 35 fragments to transmit this packet.

FW21 - 2x Destination Hdr+2x Fragment Hdr Packet containing combination of two Destination Options Headers and two Fragment Header. This test case targets extension headers processing.

FW22 - 4x Destination Hdr+3x Fragment Hdr Packet with combination of four Destination Options Headers and three Fragment Header. It targets possible flaws in processing of extension headers.

FW23 - Fragmentation "first+middle" Only the first two fragments out of three have Next Header value defined to TCP.

FW24 - Fragmentation "first(second)" Next Header of first fragment is set to ICMPv6, the second to TCP.

FW25 - Fragmentation "first#2(overlap)" First two fragments overlap. Fragments are filled with Destination Options Headers.

FW26 - Fragmentation "first#3(resend#2)" After first fragment with bogus data is sent the whole packet is then resent as an atomic fragment with the same Identification (ID).

FW27 - Fragmentation "first#4(resend#2L)" First fragment is sent and the whole packet is then resent as an atomic fragment with the same Identification (ID).

FW28 - Fragmentation "middle+last" The packet is sent in three fragments. Next Header value of the last two is set to TCP, first one to ICMPv6.

FW29 - Fragmentation "middle(first)+last" The three fragments are sent in the following order: middle, first, last. Only middle and last ones have Next Header value set to TCP. The first one is sent to ICMPv6.

FW30 - Fragmentation "last" Only the last one out of three fragments has the Next Header value set to TCP. The value of first two is set to ICMPv6.

FW31 - Plain Sending, Variable Source Ports Sends the packet in plain format but with various source ports.

7.5 Test Results

This chapter contains summarized results of all the test specified in the preceding chapter, clearly divided into sections. Subsequent chapter, Section 7.6, is all in all verbal evaluation of the appliance.

Unfortunately, unlike original intention, results of testing of only the Network IPS physical appliance will be presented herein. The underlying VMware ESXi 4.1.0 had crashed during the testing of Virtual IPS and behaved unexpectedly, therefore the results would not be reliable. The scenario and collected evidence was passed on to VMware, Inc. customer support and author of this work reserves right to not to disclose any details about this incident before a statement of the company is given.

7.5.1 IDS/IPS Assessment

Test results are summarized in Table 7.3. Detailed explanation of the results and other findigs follows.

Test Case	**Description**	**Result**
SCAN-LOC	Local Network Scanning	Failed.
SCAN-REM	Remote Network Scanning	Passed.
FL1-RA	Router Advertisement Flooding	Failed.
FL2-NS	Neighbor Solicitation Flooding	Failed.
FL3-CGA	Secure Neighbor Discovery Flooding	Failed.
RH0	Handling of Routing Header Type 0	Failed.
FRAG	Fragmentation Handling	Passed, but Exceptions Noted.
CVE	Common Vulnerabilities and Exposures	Passed, but Exceptions Noted.
FUZZ-IPv6	IPv6 Fuzzing	Passed.
FUZZ-ICMPv6	ICMPv6 Fuzzing	Passed.

Table 7.3: Network IPS General IPv6 Assessment Results

SCAN-LOC - Local Hosts Scaninng The appliance did not detect attempt to ping all nodes multicats address `FF02::1` with *ICMPv6 Echo Request* with invalid option. *ICMPv6 Parameter Problem* message was successfully received. The appliance should discover malformed packet without any known benign use. All deviations from RFC defined states should be discovered and alerted at least.

SCAN-REM - Remote Hosts Scanning This attack is actually only selective ping sweep network scan. When defined threshold is exceeded, the scan is blocked.

FL1-RA - Router Advertisement Flooding
FL2-NA - Neighbor Solicitation Flooding
FL3-CGA - Secure Neighbor Discovery Flooding

The tested appliance does not block flooding with any of these IPv6 specific messages. No more testing of flooding attacks was not necessary as reasonable assurance was obtained. This will be reported to the vendor as a suggested crucial area for improvement.

RH0 - Handling of Routing Header Type 0 The appliance let through packets containig RH0 extension header. Use of this header is depreciated due to its severe security concern and therefore there is no bening use. These packets should definitely be blocked.

FRAG - Fragmentation Handling There have been findings noted in the fragments processing. The appliance does not reassemble the whole packet but compares only n^{th} and $(n-1)^{th}$ packet to detect overlapping and only the rest of the stream is dropped after an anomaly had been found. The preceding, valid fragments are let through. This may be sufficient for detecting spot anomalies but certainly not enough to get the right context. Fragmented packets bigger than maximum MTU are let through although it may be attempt to accomplish buffer overflow. The appliance correctly drops badly chained fragments and final fragments with empty payload. However, it does not drop fragments which are smaller that minimal MTU (1280 octets) and are not the last fragment in a particular stream. Atomic fragmets, even with several Fragment Headers, are let through as well.

CVE - Common Vulnerabilities and Exposures Three out of four exploits were blocked. The one that was let through is `CVE-2003-0429`.

FUZZ-IPv6 - IPv6 Fuzzing
FUZZ-ICMPv6 - ICMPv6 Fuzzing

> The stack implementation is very robust. There was no crash or instability during the fuzzing. Moreover, several signatures reporting malformed packets triggered alerts.

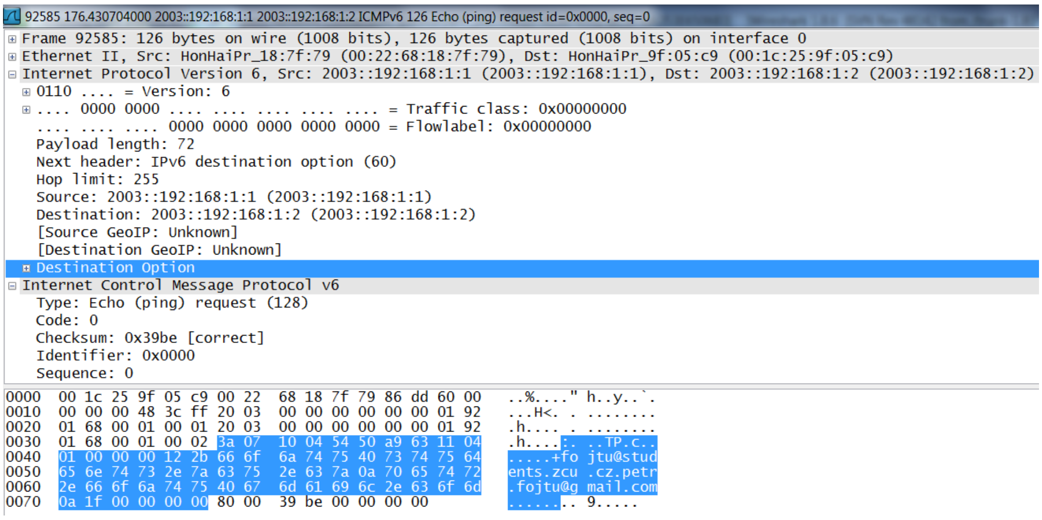
Following is a list of other findings noted during the testing.

- The appliance does not evaluate hop limit value, packets the value set to '1" are let through. This can be exploited to reflect *ICMPv6 Time Exceeded (hop limit exceeded in transit)* messages and consume resources of a router behind the IDS/IPS appliance.
- Packets with invalid checksum value are dropped but no alert is raised.
- Any packet with invalid use of Hop-by-Hop header is dropped and proper alert is raised.
- The appliance successfully detects some of the currently known firewall evasion techniques.

7.5.2 Data Loss Prevention

DLP rule looking for e-mail addresses within traffic was defined. The data can be sent using tool `covert_send6` from THC-IPv6 Attack Toolkit or by custom Scapy script. Both test cases have the same result - data were successfully transmited through the appliance.

It can be concluded that DLP inspection takes place only on higher layers of ISO/OSI model and *not* on the Layer 3. Needless to mention what the result was when encryption was employed. Figure 7.4 shows e-mail addresses hidden within the Destination Options header of ICMPv6 Echo Request message as seen in Wireshark on the target host.



```
92585 176.430704000 2003::192:168:1:1 2003::192:168:1:2 ICMPv6 126 Echo (ping) request id=0x0000, seq=0
⊞ Frame 92585: 126 bytes on wire (1008 bits), 126 bytes captured (1008 bits) on interface 0
⊞ Ethernet II, Src: HonHaiPr_18:7f:79 (00:22:68:18:7f:79), Dst: HonHaiPr_9f:05:c9 (00:1c:25:9f:05:c9)
⊟ Internet Protocol Version 6, Src: 2003::192:168:1:1 (2003::192:168:1:1), Dst: 2003::192:168:1:2 (2003::192:168:1:2)
  ⊞ 0110 .... = Version: 6
  ⊞ .... 0000 0000 .... .... .... .... .... = Traffic class: 0x00000000
    .... .... .... 0000 0000 0000 0000 0000 = Flowlabel: 0x00000000
    Payload length: 72
    Next header: IPv6 destination option (60)
    Hop limit: 255
    Source: 2003::192:168:1:1 (2003::192:168:1:1)
    Destination: 2003::192:168:1:2 (2003::192:168:1:2)
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
  ⊞ Destination Option
⊟ Internet Control Message Protocol v6
    Type: Echo (ping) request (128)
    Code: 0
    Checksum: 0x39be [correct]
    Identifier: 0x0000
    Sequence: 0
0000  00 1c 25 9f 05 c9 00 22  68 18 7f 79 86 dd 60 00   ..%....." h..y..`.
0010  00 00 00 48 3c ff 20 03  00 00 00 00 00 00 01 92   ...H<. . ........
0020  01 68 00 01 00 01 20 03  00 00 00 00 00 00 01 92   .h.... . ........
0030  01 68 00 01 00 02 3a 07  10 04 54 50 a9 63 11 04   .h....:. ..TP.c..
0040  01 00 00 00 12 2b 66 6f  6a 74 75 40 73 74 75 64   .....+fo jtu@stud
0050  65 6e 74 73 2e 7a 63 75  2e 63 7a 0a 70 65 74 72   ents.zcu .cz.petr
0060  2e 66 6f 6a 74 75 40 67  6d 61 69 6c 2e 63 6f 6d   .fojtu@g mail.com
0070  0a 1f 00 00 00 00 80 00  39 be 00 00 00 00         ........ 9.....
```

Figure 7.4: E-mail Addresses Hidden Within Destination Options Header

7.5.3 Firewall

Testing of the appliance was performed in two rounds. During the first round, IDS/IPS IPv6 signatures were enabled whilst during the second round were deactivated. The results are summarized in Table 7.4.

The results are divided into two columns, one for each round. *Blocked* means that the packet was not allowed through the appliance, *Let Through* then means that it traversed the appliance. In all cases when the packet passed the appliance, TCP `SYN-ACK` packet was received by the source host.

It is obvious that bypass the firewall does not require much effort. Unfortunately, the situation is not very different among other vendors [21] [22] [26] and it is necessary to mention that the firewall is not a primary function of the appliance. However, it is interesting, if not shocking, that the firewall is vulnerable to the same evasion techniques that are detectable by the IDS/IPS engine. Concretely, test cases `FW16`, `FW17`, `FW21` and `FW22` triggered alert through signature `Firewall_Bypass`.

Test	**Description**	**Firewall&IPS**	**Firewall Only**
FW1	Plain Sending	Blocked	Blocked
FW2	Plain Sending with Data	Blocked	Blocked
FW3	IPv4 Ethernet Type	Blocked	Blocked
FW4	Hop-by-Hop Hdr, Ignore Option	Blocked	Blocked
FW5	Destination Hdr, Ignore Option	Blocked	Blocked
FW6	Hop-by-Hop Hdr, Router Alert	Blocked	Blocked
FW7	3x Destination Hdr, Ignore Option	Blocked	Blocked
FW8	130x Destination Hdr, Ignore Option	Blocked	Blocked
FW9	Atomic Fragment	Let Through	Let Through
FW10	2x Atomic Fragment, Same ID	Let Through	Let Through
FW11	2x Atomic Fragment, Different ID	Let Through	Let Through
FW12	3x Atomic Fragment, Same ID	Let Through	Let Through
FW13	3x Atomic Fragment, Different ID	Let Through	Let Through
FW14	130x Atomic Fragment, Same ID	Let Through	Let Through
FW15	130x Atomic Fragment, Different ID	Let Through	Let Through
FW16	260x Atomic Fragment, Same ID	Blocked	Let Through
FW17	260x Atomic Fragment, Different ID	Blocked	Let Through
FW18	2KB Destination Hdr	Blocked	Blocked
FW19	2KB Destination Hdr+Destination Hdr	Blocked	Blocked
FW20	32x 2KB Destination Hdr	Blocked	Blocked
FW21	2x Destination Hdr+2x Fragment Hdr	Blocked	Let Through
FW22	4x Destination Hdr+3x Fragment Hdr	Blocked	Let Through
FW23	Fragmentation “first+middle”	Blocked	Blocked
FW24	Fragmentation “first(second)”	Blocked	Blocked
FW25	Fragmentation “first#2(overlap)”	Blocked	Blocked
FW26	Fragmentation “first#3(resend#2)”	Let Through	Let Through
FW27	Fragmentation “first#4(resend#2L)”	Blocked	Blocked
FW28	Fragmentation “middle+last”	Blocked	Blocked
FW29	Fragmentation “middle(first)+last”	Blocked	Blocked
FW30	Fragmentation “last”	Blocked	Blocked
FW31	Plain Sending, Variable Source Ports	Blocked	Blocked

Table 7.4: Network IPS Firewall Assessment Results

7.6 Assessment Conclusion

The overall results as presented in previous chapters are not very satisfactory. This chapter will discuss other aspects of tested appliance and suggest ways to achieve higher level of protection. It is necessary to mention that there has been no market-driven demand for complex IPv6 support so far in general and this work does not intend to cast a bad light on the tested appliance.

Network IPS as a general system of active protection is positioned into vendors portfolio where it supports other products so they can create complex security solutions together. Its versatility, on the other hand, may represent a certain disadvantage in highly specialized areas. The appliance supports variety of protocols spanning several areas of networking and provides protection against numerous threats but it is specificaly focused on threats linked with application layer. These threats endanger aplication, web and database servers where, in extreme case, one maliciously crafted packet can cause harm. Without an appropriate protection, these attacks are often discovered when it is too late. Network IPS provides an early protection against this type of attacks and its detection engine utilizes algorithms rather then signatures to protect against all possible attack modifications where simple search for patterns would be inadequate.

Valuable functionality of the appliance is protection against worms. It well accompanies antimalware software on hosts because network worms are usually not planted in a file system and therefore are undetectable by common host-based solutions. Network IPS is more than appropriate protection againts worm-like malware.

Several attacks may be detectable by flow analysis. Version 0.0 of Network IPS firmware can generate flow reports. However, it can not analyze the reports but rather export them to external collectors. Vendor offers a complemetary solution to IDS/IPS appliances, Flow Collector, which posses such functionality. With advanced flow analysis, both solutions together can provide high level of protection againts flooding attacks, even the skillfully modified ones.

Another interesting fact is that vendor collects real data feedback through its more than 0000 customers worldwide. Vendors security research team then has means to search for and protect against zero-day exploits and they have the best and undistorted testing environment - real world data.

8 Epilogue

The goal of this work was to gather knowledge of IPv6 security and related threats, then look into this area from perspective of current IDS/IPS solutions and afterwards transform the gained knowledge into practical guidelines how to assess usability of these systems.

The first part of this work contains comprehensive and up-to-date comparison of IPv4 and IPv6 related threats with references to corresponding RFCs. This part may be useful as a reference for future work. However, any such potential work should take into account that IPv6 is very dynamic and still developing technology. In fact, some of the information may become outdated in a couple of months.

The second part focused on particular attacks and IDS/IPS appliance assessment. I see the main contribution of this work in desciption of the selected attacks. Eventhough several ready-to-use tools for penetration testing exist, none of them comes with any kind of documentation. Original intention was to test physical and virtual appliance with same firmware and compare performace results. However, issue in the VMware virtual infrastructure was found during the testing so I decided, after consultation with the thesis supervisor, to scratch the results as untrustworthy. Testing of additional functionalities of the physical appliance was performed as a subtitute.

The overall results of the assessment are unsatisfactory. It is necessary to mention that the situation among the majority of other vendors is very similar. I strongly believe that such testing will help to improve IPv6 capabilities and hopefully even the protocol itself. There is a wide range of possibilities for future work as well as challenges in the area of IPv6 security. The most current one would be transition mechanisms from IPv4 to IPv6 and its coexistence. Further development of testing tools and testc cases would be advisable as well.

In conclusion, it can not be decided whether IPv6 is by design more secure than IPv4. It is just different, maybe more different than many expected. Wider deployment or testing of IPv6 capable solutions in real-world scenarios would help to break the barrier and market demand would drive vendors to provide better IPv6 support and consequently the technology would be more trusted.

A List of Abbreviations

ACL	Access List
AH	Authentication Header
ARP	Address Resolution Protocol
BGP	Border Gateway Protocol
BU	Binding Update
CGA	Cryptographically Generated Addresses
CN	Correspondent Node
CoA	Care-of Address
CPU	Central Processing Unit
DAD	Duplicate Address Detection
DLP	Data Loss Prevention
DoS	Denial of Service
DDoS	Distributed Denial of Service
DHCP	Dynamic Host Configuration Protocol
DMZ	Demilitarized Zone
DNS	Domain Name System
DoS	Denial of Services
DST	Destination
ESP	Encapsulating Security Payload
HA	Home Agent
HDR	Header
HIDS	Host Intrusion Detection System
HIPS	Host Intrusion Prevention System
HMAC	Hash Message Authentication Code
HoA	Home Address
IANA	Internet Assigned Numbers Authority
ICMP	Internet Control Message Protocol
ICMPv6	Internet Control Message Protocol version 6
IDS	Intrusion Detection System
IETF	Internet Engineering Task Force
IP	Internet Protocol
IPS	Intrusion Prevention System
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
IS-IS	Intermediate System-to-Intermediate System
ISIC	IP Stack Integrity Checker

ISO/OSI	International Standards Organization/Open Systems Interconnection
ISP	Internet Service Provider
JRE	Java Runtime Environment
LAN	Local Area Network
LMI	Local Management Interface
MD5	Message-Digest Algorithm 5
MIPv6	Mobile Internet Protocol version 6
MITM	Man-in-the-middle
MLD	Multicast Listener Discovery
MN	Mobile Node
MTU	Maximum Transmission Unit
NA	Neighbor Advertisment
NAT	Network Address Translation
NDP	Neighbor Discovery Protocol
NUD	Neighbor Unreachability Detection
NIDS	Network Intrusion Detection System
NIPS	Network Intrusion Prevention System
NS	Neighbor Solicitation
OS	Operating System
OSPFv3	Open Shortest Path First version 3
PMTU	Path Maximum Transmission Unit
PMTUD	Path Maximum Transmission Unit Discovery
QoS	Quality of Services
RA	Router Advertisement
RFC	Request for Comments
RH	Routing Header
RH0	Routing Header Type 0
RIPng	Routing Information Protocol Next-Generation
RSA	Rivest, Shamir and Adleman Algorithm
RSVP	Resources Reservation Protocol
SA	Security Association
SEND	Secure Neighbor Discovery
SLAAC	Stateless Address Autoconfiguration
SRC	Source
SSH	Secure Shell
TC	Test Case
TCP	Transmission Control Protocol
THC	The Hacker's Choice
TLV	Type-Length-Value
UDP	User Datagram Protocol
WAP	Wireless Access Point

B Bibliography and Sources

[1] PILIHANTO, Atik. A Complete Guide on IPv6 Attack and Defense. Advisor: Rick Wanner. SANS Institute InfoSec Reading Room, 2012.

[2] HOGG, Scott and VYNCKE, Eric. IPv6 Security: Protection Measures for the Next Internet Protocol. Cisco Press, 2011. ISBN 978-1-58705-594-2.

[3]SATRAPA, Pavel. Internetový protokol verze 6. CZ.NIC, 2011. ISBN 978-80-904248-4-5.

[4]CONVERY, Sean and MILLER, Darrin. IPv6 and IPv4 Threat Comparison and Best-Practice Evaluation (v1.0). Cisco Systems Technical Report, 2004.

[5]FRANKEL, Sheila, GRAVEMAN, Richard, PEARCE John and ROOKS, Mark. Guidelines for the Secure Deployment of IPv6: Recommendations of the National Institute of Standards and Technology. NIST, Special Publication 800-119, 2010.

[6]KIBIRKSTIS, Algis. Intrusion Detection FAQ: What Are The Top Selling IDS/IPS and What Differentiates Them from Each Other? [online]. SANS Institute, November 2009. Accessed March 2013 at <http://www.sans.org/security-resources/idfaq/top-selling-ids-ips.php>

[7] IBM Proventia Network IPS Firmware Version 4.1 [online]. IBM Internet Security Systems, 2010. Retrieved March 2013 from <http://pic.dhe.ibm.com/infocenter/sprotect/v2r8m0/topic/com.ibm.ips.doc/pdfs/ProventiaIPS_System_Requirements.pdf>

[8] HP N Platform Next-Generation Intrusion Prevention System (NGIPS) Data Sheet [online]. Hewlett-Packard, 2012. Retrieved March 2013 from <http://h20195.www2.hp.com/V2/GetDocument.aspx?docname=4AA3-0819ENW&cc=us&lc=en>

[9] IDP Series Feature Documentation [online]. Juniper Networks, 2012. Retrieved March 2013 from <http://www.juniper.net/techpubs/en_US/idp5.1/information-products/pathway-pages/features-index.pdf>

[10] McAfee Network Security Platform Datasheet [online]. McAfee, 2013. Retrieved March 2013 from <http://www.mcafee.com/uk/resources/data-sheets/ds-network-security-platform-m-series.pdf>

[11] Sourcefire Next-Generation IPS [online]. Sourcefire, 2013. Accessed March 2013 at <https://na8.salesforce.com/sfc/p/80000000dRH9IX6H8ZNi6bV0SOcfM9mwoKFSoZU=>

[12] Snort 2.8.0 new features: IPv6 and port lists [online]. TechTarget, 2007. Accessed March 2013 at <http://searchitchannel.techtarget.com/tip/Snort-280-new-features-IPv6-and-port-lists>

[13] Check Point Software Technologies Website, IPv6 and Intrusion Prevention: What You Need To Know [online]. Security Café Reading Room, 2011. Accessed April 2013 at <http://www.checkpoint.com/securitycafe/readingroom/intrusion/ipv6_intrusion_prevention.html>

[14] Cisco Systems Solutions Website, IPv6 [online]. Cisco Systems Inc. Accessed April 2013 at <http://www.cisco.com/web/solutions/trends/ipv6/index.html>

[15] IPv6 Multicast Address Space Registry [online]. Internet Assigned Numbers Authority, 2013. Accessed March 2013 at <http://www.iana.org/assignments/ipv6-multicast-addresses/ipv6-multicast-addresses.xml>

[16] Protocol Numbers [online]. Internet Assigned Numbers Authority, 2013. Accessed March 2013 at <http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xml>

[17] ULLRICH, Johannes. IPv6 Focus Month: Kaspersky Firewall IPv6 Vulnerability [online]. Internet Storm Center, March 2013. Accessed April 2013 at <http://isc.sans.edu/diary/IPv6+Focus+Month%3A+Kaspersky+Firewall+IPv6+Vulnerability/15397>

[18] SI6 Networks Website, SI6 Networks' IPv6 Toolkit: A security assessment and troubleshooting tool for the IPv6 protocols [online]. SI6 Networks, 2013. Accessed April 2013 at <http://si6networks.com/tools/ipv6toolkit/>

[19] ISIC Project Website, ISIC – IP Stack Integrity Checker [online]. Accessed April 2013 at <http://isic.sourceforge.net/>

[20] Kali Linux Documentation [online]. Offensive Security, 2013. Accessed April 2013 at <http://www.kali.org/official-documentation/>

[21] HEUSE, Marc. IPv6 Insecurity Revolutions. Hack in the Box, October 8 - 11, 2012. Kuala Lumpur, Malaysia.

[22] HEUSE, Marc. IPv6 Vulnerabilities, Failures and a Future? [online]. Marc Heuse's Personal Webpage, November 2011. Accessed May 2013 at <http://www.mh-sec.de/downloads/mh-ipv6_vulnerabilities.pdf>

[23] GONT, Fernando. Results of a Security Assessment of the Internet Protocol version 6 (IPv6). Hack in Paris, June 18 - 22, 2012. Paris, France.

[24] GONT, Fernando. Recent Advances in IPv6 Security. Hackers to Hackers Conference, October 20 - 21, 2012. Sao Paulo, Brazil.

[25] ATLASIS, Antonios. Attacking IPv6 Implementation Using Fragmentation. Black Hat Europe, March 14 - 15 2012. Amsterdam, Netherlands.

[26] GONT, Fernando and HEUSE, Marc. Security Assessments of IPv6 Networks and Firewalls. IPv6 Kongress 2013, June 6 - 7. Frankfurt, Germany.

[27] PIVARNÍK, Jozef and GRÉGR, Matěj. Rogue Router Advertisement Attack [online]. Vysoké učení technické v Brně, May 2013. Accessed May 2013 at <http://6lab.cz/article/rogue-router-advertisement-attack/>

[*RFC* 1752] BRANDNER, S. and MANKIN, A. RFC 1752, "The Recommendation for the IP Next Generation Protocol". January 1995.

[*RFC* 1981] McCANN, J., DEERING, S. and MOGUL, J. RFC 1981, "Path MTU Discovery for IP version 6". August 1996.

[*RFC* 2385] HEFFERNAN, A. RFC 2385, “Protection of BGP Sessions via the TCP MD5 Signature Option”. August 1998.

[*RFC* 2460] DEERING, S. and HINDEN, R. RFC 2460, “Internet Protocol, Version 6 (IPv6) Specification”. December 1998.

[*RFC* 2675] BORMAN, D., DEERING, S. and HINDEN, R. RFC 2675, “IPv6 Jumbograms”. August 1999.

[*RFC* 2711] PARTRIDGE, C., JACKSON, A. RFC 2711, “IPv6 Router Alert Option”. October 1999.

[*RFC* 2827] FERGUSON, P. and SENIE, D. RFC 2827, “Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing”. May 2000.

[*RFC* 3315] DROMS, R., BOUND, J., VOLZ, B., LEMON, T., PERKINS, C. and CARNEY, M. RFC 3315, “Dynamic Host Configuration Protocol for IPv6 (DHCPv6)”. July 2003.

[*RFC* 3567] LI, T. and ATKINSON, R. RFC 3567, “Intermediate System to Intermediate System (IS-IS) Cryptographic Authentication”. July 2003.

[*RFC* 3697] RAJAHALME, J., CONTA, A., CARPENTER, B. and DEERING, S. RFC 3697, “IPv6 Flow Label Specification”. March 2004.

[*RFC* 3756] NIKANDER, P., KEMPF, J. and NORDMARK, E. RFC 3756, “IPv6 Neighbor Discovery (ND) Trust Models and Threats”. May 2004.

[*RFC* 3927] CHESHIRE, S., ABOBA, B., GUTTMAN, E. RFC 3927, “Dynamic Configuration of IPv4 Link-Local Addresses”. May 2005.

[*RFC* 3971] ARKKO, J., KEMPF, J., ZILL, B. and NIKANDER, P. RFC 3971, “SEcure Neighbor Discovery (SEND)”. March 2005.

[*RFC* 3972] AURA, T. RFC 3972, “Cryptographically Generated Addresses (CGA)”. March 2005.

[*RFC* 4301] KENT, S. and SEO, K. RFC 4301, "Security Architecture for the Internet Protocol". December 2005.

[*RFC* 4302] KENT, S. RFC 4302, "IP Authentication Header". December 2005.

[*RFC* 4303] KENT, S. RFC 4302, "IP Encapsulating Security Payload (ESP)". December 2005.

[*RFC* 4443] CONTA, A., DERING, S. and GUPTA, M. RFC 4443, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification". March 2006.

[*RFC* 4861] NARTEN, T., NORDMARK, E., SIMPSON, W., SOLIMAN, H. RFC 4861, "Neighbor Discovery for IP version 6 (IPv6)". September 2007.

[*RFC* 4862] THOMSON, S., NARTEN, T. and JINMEI, T. RFC 4862, "IPv6 Stateless Address Autoconfiguration". September 2007.

[*RFC* 4890] DAVIES, E. and MOHACSI, J. RFC 4890, "Recommendations for Filtering ICMPv6 Messages in Firewalls". May 2007.

[*RFC* 5095] ABLEY, J., SAVOLA, P. and NEVILLE-NEIL, G. RFC 5095, "Deprecation of Type 0 Routing Headers in IPv6". December 2007.

[*RFC* 5722] KRISHNAN, S. RFC 5722, "Handling of Overlapping IPv6 Fragments". December 2009.

[*RFC* 5996] KAUFMAN, C., HOFFMAN, P., NIR, Y., ERONEN, P. RFC 5996, "Internet Key Exchange Protocol Version 2 (IKEv2)". September 2010.

[*RFC* 6275] PERKINS, C., JOHNSON, D. and ARKKO, J. RFC 6275, "Mobility Support in IPv6". July 2011.

[*RFC* 6946] GONT, F. RFC 6949, "Processing of IPv6 "Atomic" Fragments". May 2013.

[*RFC DRAFT*1] GONT, F and CHOWN, T. draft-ietf-opsec-ipv6-host-scanning-01, "Network Reconnaissance in IPv6 Networks". 30 April 2013.

C Attachment I - Multicasts

Node-, Link- and Site-local multicast addresses registered by IANA [15] as of February 2013.

Node-Local Scope Multicast Addresses

Address	Description
FF01:0:0:0:0:0:0:1	All Nodes Address
FF01:0:0:0:0:0:0:2	All Routers Address
FF01:0:0:0:0:0:0:FB	mDNSv6

Link-Local Scope Multicast Addresses

Address	Description
FF02:0:0:0:0:0:0:1	All Nodes Address
FF02:0:0:0:0:0:0:2	All Routers Address
FF02:0:0:0:0:0:0:3	Unassigned
FF02:0:0:0:0:0:0:4	DVMRP Routers
FF02:0:0:0:0:0:0:5	OSPFIGP
FF02:0:0:0:0:0:0:6	OSPFIGP Designated Routers
FF02:0:0:0:0:0:0:7	ST Routers
FF02:0:0:0:0:0:0:8	ST Hosts
FF02:0:0:0:0:0:0:9	RIP Routers
FF02:0:0:0:0:0:0:A	EIGRP Routers
FF02:0:0:0:0:0:0:B	Mobile-Agents
FF02:0:0:0:0:0:0:C	SSDP
FF02:0:0:0:0:0:0:D	All PIM Routers
FF02:0:0:0:0:0:0:E	RSVP-ENCAPSULATION
FF02:0:0:0:0:0:0:F	UPnP
FF02:0:0:0:0:0:0:10	All-BBF-Access-Nodes
FF02:0:0:0:0:0:0:12	VRRP
FF02:0:0:0:0:0:0:16	All MLDv2-capable routers
FF02:0:0:0:0:0:0:1A	all-RPL-nodes

Address	**Description**
FF02:0:0:0:0:0:0:6A	All-Snoopers
FF02:0:0:0:0:0:0:6B	PTP-pdelay
FF02:0:0:0:0:0:0:6C	Saratoga
FF02:0:0:0:0:0:0:6D	LL-MANET-Routers
FF02:0:0:0:0:0:0:6E	IGRS
FF02:0:0:0:0:0:0:6F	iADT Discovery
FF02:0:0:0:0:0:0:FB	mDNSv6
FF02:0:0:0:0:0:1:1	Link Name
FF02:0:0:0:0:0:1:2	All-dhcp-agents
FF02:0:0:0:0:0:1:3	Link-local Multicast Name Resolution
FF02:0:0:0:0:0:1:4	DTCP Announcement
FF02:0:0:0:0:0:1:5	afore_vdp
FF02:0:0:0:0:0:1:6	Babel
FF02::1:FF00:0000/104	Solicited-Node Address
FF02:0:0:0:0:2:FF00::/104	Node Information Queries

Site-Local Scope Multicast Addresses

FF05:0:0:0:0:0:0:2	All Routers Address
FF05:0:0:0:0:0:0:FB	mDNSv6
FF05:0:0:0:0:0:1:3	All-dhcp-servers
FF05:0:0:0:0:0:1:4	Deprecated (2003-03-12)
FF05:0:0:0:0:0:1:5	SL-MANET-ROUTERS

D Attachment II - ICMPv6 Filtering

ICMPv6 filtering recommendations as defined in [*RFC* 4890].

Must - must be dropped; **Must Not** - must not be dropped; **Policy Defined** - policy should be defined; **Should** - should be dropped unless a good case can be made; **Should Not** - normally should not be dropped; **Will** - will be dropped anyway, no special attention needed.

Type	Message / Description	Transit Traffic	Local Traffic
1	Destination Unreachable	Must Not	Must Not
2	Packet Too Big	Must Not	Must Not
3	Time Exceeded	Must Not	Must Not
3-Code 1	Time Exceeded	Should Not	Should Not
4	Parameter Problem	Must Not	Must Not
4-Code 0	Parameter Problem	Should Not	Should Not
5 - 99	Unallocated Error messages	Policy Defined	Policy Defined
100 - 101	Experiment	Should	Should
102 - 126	Unallocated Error messages	Policy Defined	Policy Defined
127	Extension	Should	Should
128	Echo Request	Must Not	Must Not
129	Echo Response	Must Not	Must Not
130	Listener Query	Will	Must Not
131	Listener Report	Will	Must Not
132	Listener Done	Will	Must Not
133	Router Solicitation	Will	Must Not
134	Router Advertisement	Will	Must Not
135	Neighbor Solicitation	Will	Must Not
136	Neighbor Advertisement	Will	Must Not
137	Redirect	Will	Policy Defined
138	Router Renumbering	Should	Will
139	Node Information Query	Should	Policy Defined
140	Node Information Response	Should	Policy Defined
141	Inverse ND Solicitation	Will	Must Not
142	Inverse ND Advertisement	Will	Must Not
143	Listener Report v2	Will	Must Not
144	HA Address Discovery Request	Should Not	Will
145	HA Address Discovery Reply	Should Not	Will

Type	**Message / Description**	**Transit Traffic**	**Local Traffic**
146	Mobile Prefix Solicitation	Should Not	Will
147	Mobile Prefix Advertisement	Should Not	Will
148	Certificate Path Solicitation	Will	Must Not
149	Certificate Path Advertisement	Will	Must Not
150	Seamoby Experimental	Policy Defined	Will
151	Multicast Router Advertisement	Will	Must Not
152	Multicast Router Solicitation	Will	Must Not
153	Multicast Router Termination	Will	Must Not
154 - 199	Informative	Policy Defined	Should
200 - 201	Experimental	Should	Should
202 - 254	Informative	Policy Defined	Should
255	Extension	Should	Should

E Attachment III - Next Header

Protocol numbers used in IPv6 header field called the “Next Header” field registed by IANA [16] as of February 2013.

Decimal	Keyword	Protocol
0	HOPOPT	IPv6 Hop-by-Hop Option
1	ICMP	Internet Control Message
2	IGMP	Internet Group Management
3	GGP	Gateway-to-Gateway
4	IPv4	IPv4 encapsulation
5	ST	Stream
6	TCP	Transmission Control
7	CBT	CBT
8	EGP	Exterior Gateway Protocol
9	IGP	any private interior gateway (used by Cisco for their IGRP)
10	BBN-RCC-MON	BBN RCC Monitoring
11	NVP-II	Network Voice Protocol
12	PUP	PUP
13	ARGUS	ARGUS
14	EMCON	EMCON
15	XNET	Cross Net Debugger
16	CHAOS	Chaos
17	UDP	User Datagram
18	MUX	Multiplexing
19	DCN-MEAS	DCN Measurement Subsystems
20	HMP	Host Monitoring
21	PRM	Packet Radio Measurement
22	XNS-IDP	XEROX NS IDP
23	TRUNK-1	Trunk-1
24	TRUNK-2	Trunk-2
25	LEAF-1	Leaf-1
26	LEAF-2	Leaf-2
27	RDP	Reliable Data Protocol
28	IRTP	Internet Reliable Transaction
29	ISO-TP4	ISO Transport Protocol Class 4
30	NETBLT	Bulk Data Transfer Protocol
31	MFE-NSP	MFE Network Services Protocol

Decimal	**Keyword**	**Protocol**
32	MERIT-INP	MERIT Internodal Protocol
33	DCCP	Datagram Congestion Control Protocol
34	3PC	Third Party Connect Protocol
35	IDPR	Inter-Domain Policy Routing Protocol
36	XTP	XTP
37	DDP	Datagram Delivery Protocol
38	IDPR-CMTP	IDPR Control Message Transport Proto
39	TP++	TP++ Transport Protocol
40	IL	IL Transport Protocol
41	IPv6	IPv6 encapsulation
42	SDRP	Source Demand Routing Protocol
43	IPv6-Route	Routing Header for IPv6
44	IPv6-Frag	Fragment Header for IPv6
45	IDRP	Inter-Domain Routing Protocol
46	RSVP	Reservation Protocol
47	GRE	Generic Routing Encapsulation
48	DSR	Dynamic Source Routing Protocol
49	BNA	BNA
50	ESP	Encap Security Payload
51	AH	Authentication Header
52	I-NLSP	Integrated Net Layer Security TUBA
53	SWIPE	IP with Encryption
54	NARP	NBMA Address Resolution Protocol
55	MOBILE	IP Mobility
56	TLSP	Transport Layer Security Protocol using Kryptonet key management
57	SKIP	SKIP
58	IPv6-ICMP	ICMP for IPv6
59	IPv6-NoNxt	No Next Header for IPv6
60	IPv6-Opts	Destination Options for IPv6
61		any host internal protocol
62	CFTP	CFTP
63		any local network
64	SAT-EXPAK	SATNET and Backroom EXPAK
65	KRYPTOLAN	Kryptolan
66	RVD	MIT Remote Virtual Disk Protocol
67	IPPC	Internet Pluribus Packet Core
68		any distributed file system
69	SAT-MON	SATNET Monitoring

Decimal	**Keyword**	**Protocol**
70	VISA	VISA Protocol
71	IPCV	Internet Packet Core Utility
72	CPNX	Computer Protocol Network Executive
73	CPHB	Computer Protocol Heart Beat
74	WSN	Wang Span Network
75	PVP	Packet Video Protocol
76	BR-SAT-MON	Backroom SATNET Monitoring
77	SUN-ND	SUN ND PROTOCOL-Temporary
78	WB-MON	WIDEBAND Monitoring
79	WB-EXPAK	WIDEBAND EXPAK
80	ISO-IP	ISO Internet Protocol
81	VMTP	VMTP
82	SECURE-VMTP	SECURE-VMTP
83	VINES	VINES
84	TTP	TTP
84	IPTM	Protocol Internet Protocol Traffic Manager
85	NSFNET-IGP	NSFNET-IGP
86	DGP	Dissimilar Gateway Protocol
87	TCF	TCF
88	EIGRP	EIGRP
89	OSPFIGP	OSPFIGP
90	Sprite-RPC	Sprite RPC Protocol
91	LARP	Locus Address Resolution Protocol
92	MTP	Multicast Transport Protocol
93	AX.25	AX.25 Frames
94	IPIP	IP-within-IP Encapsulation Protocol
95	MICP	Mobile Internetworking Control Pro.
96	SCC-SP	Semaphore Communications Sec. Pro.
97	ETHERIP	Ethernet-within-IP Encapsulation
98	ENCAP	Encapsulation Header
99		any private encryption scheme
100	GMTP	GMTP
101	IFMP	Ipsilon Flow Management Protocol
102	PNNI	PNNI over IP
103	PIM	Protocol Independent Multicast
104	ARIS	ARIS
105	SCPS	SCPS
106	QNX	QNX
107	A/N	Active Networks

Decimal	**Keyword**	**Protocol**
108	IPComp	IP Payload Compression Protocol
109	SNP	Sitara Networks Protocol
110	Compaq-Peer	Compaq Peer Protocol
111	IPX-in-IP	IPX in IP
112	VRRP	Virtual Router Redundancy Protocol
113	PGM	PGM Reliable Transport Protocol
114		any 0-hop protocol
115	L2TP	Layer Two Tunneling Protocol
116	DDX	D-II Data Exchange (DDX)
117	IATP	Interactive Agent Transfer Protocol
118	STP	Schedule Transfer Protocol
119	SRP	SpectraLink Radio Protocol
120	UTI	UTI
121	SMP	Simple Message Protocol
122	SM	SM
123	PTP	Performance Transparency Protocol
124	ISIS over IPv4	
125	FIRE	
126	CRTP	Combat Radio Transport Protocol
127	CRUDP	Combat Radio User Datagram
128	SSCOPMCE	
129	IPLT	
130	SPS	Secure Packet Shield
131	PIPE	Private IP Encapsulation within IP
132	SCTP	Stream Control Transmission Protocol
133	FC	Fibre Channel
134	RSVP-E2E-IGNORE	
135	Mobility Header	
136	UDPLite	
137	MPLS-in-IP	
138	manet	MANET Protocols
139	HIP	Host Identity Protocol
140	Shim6	Shim6 Protocol
141	WESP	Wrapped Encapsulating Security Payload
142	ROHC	Robust Header Compression
143-252		Unassigned
253		Use for experimentation and testing
254		Use for experimentation and testing
255	Reserved	