

IPv6 Operations Working Group (v6ops)  
Internet-Draft  
Updates: [6105](#) (if approved)  
Intended status: Informational  
Expires: May 18, 2013

F. Gont  
UK CPNI  
November 14, 2012

**Implementation Advice for IPv6 Router Advertisement Guard (RA-Guard)**  
**draft-ietf-v6ops-ra-guard-implementation-06**

Abstract

The IPv6 Router Advertisement Guard (RA-Guard) mechanism is commonly employed to mitigate attack vectors based on forged ICMPv6 Router Advertisement messages. Many existing IPv6 deployments rely on RA-Guard as the first line of defense against the aforementioned attack vectors. However, some implementations of RA-Guard have been found to be prone to circumvention by employing IPv6 Extension Headers. This document describes the evasion techniques that affect the aforementioned implementations, and formally updates [RFC 6105](#), such that the aforementioned RA-Guard evasion vectors are eliminated.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

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

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

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

Copyright Notice

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

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

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

## Table of Contents

<a href="#">1.</a>	Introduction . . . . .	<a href="#">3</a>
<a href="#">2.</a>	Evasion techniques for some Router Advertisement Guard (RA Guard) implementations . . . . .	<a href="#">4</a>
<a href="#">2.1.</a>	Attack Vector based on IPv6 Extension Headers . . . . .	<a href="#">4</a>
<a href="#">2.2.</a>	Attack vector based on IPv6 fragmentation . . . . .	<a href="#">4</a>
<a href="#">3.</a>	RA-Guard implementation advice . . . . .	<a href="#">8</a>
<a href="#">4.</a>	Other Implications . . . . .	<a href="#">11</a>
<a href="#">5.</a>	IANA Considerations . . . . .	<a href="#">12</a>
<a href="#">6.</a>	Security Considerations . . . . .	<a href="#">13</a>
<a href="#">7.</a>	Acknowledgements . . . . .	<a href="#">14</a>
<a href="#">8.</a>	References . . . . .	<a href="#">15</a>
<a href="#">8.1.</a>	Normative References . . . . .	<a href="#">15</a>
<a href="#">8.2.</a>	Informative References . . . . .	<a href="#">15</a>
<a href="#">Appendix A.</a>	Assessment tools . . . . .	<a href="#">17</a>
	Author's Address . . . . .	<a href="#">18</a>



## **1. Introduction**

IPv6 Router Advertisement Guard (RA-Guard) is a mitigation technique for attack vectors based on ICMPv6 Router Advertisement messages. [RFC6104] describes the problem statement of "Rogue IPv6 Router Advertisements", and [RFC6105] specifies the "IPv6 Router Advertisement Guard" functionality.

The concept behind RA-Guard is that a layer-2 device filters ICMPv6 Router Advertisement messages, according to a number of different criteria. The most basic filtering criterion is that Router Advertisement messages are discarded by the layer-2 device unless they are received on a specified port of the layer-2 device. Clearly, the effectiveness of the RA Guard mitigation relies on the ability of the layer-2 device to identify ICMPv6 Router Advertisement messages.

Some popular RA-Guard implementations have been found to be easy to circumvent by employing IPv6 extension headers [CPNI-IPv6]. This document describes such evasion techniques, and provides advice to RA-Guard implementers such that the aforementioned evasion vectors can be eliminated.

It should be noted that the aforementioned techniques could also be exploited to evade network monitoring tools such as NDPMon [NDPMon], ramond [ramond], and rafixd [rafixd], and could probably be exploited to perform stealth DHCPv6 attacks.

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



## **2. Evasion techniques for some Router Advertisement Guard (RA Guard) implementations**

The following subsections describe two different vectors that have been found to be effective for the evasion of popular implementations of the RA-Guard protection. [Section 2.1](#) describes an attack vector based on the use of IPv6 Extension Headers with the ICMPv6 Router Advertisement messages, which may be used to circumvent the RA-Guard protection of those implementations that fail to process an entire IPv6 header chain when trying to identify the ICMPv6 Router Advertisement messages. [Section 2.2](#) describes an attack method based on the use of IPv6 fragmentation, possibly in conjunction with the use of IPv6 Extension Headers. This later vector has been found to be effective with all existing implementations of the RA-Guard mechanism.

### **2.1. Attack Vector based on IPv6 Extension Headers**

While there is currently no legitimate use for IPv6 Extension Headers in ICMPv6 Router Advertisement messages, Neighbor Discovery implementations allow the use of Extension Headers with these messages, by simply ignoring the received options. Some RA-Guard implementations try to identify ICMPv6 Router Advertisement messages by simply looking at the "Next Header" field of the fixed IPv6 header, rather than following the entire header chain. As a result, such implementations fail to identify any ICMPv6 Router Advertisement messages that include any Extension Headers (for example, a Hop by Hop Options header, a Destination Options Header, etc.), and can be easily circumvented.

The following figure illustrates the structure of ICMPv6 Router Advertisement messages that implement this evasion technique:

```

+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|NH=60|      |NH=58|      |                                     |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
| IPv6 header | Dst Opt Hdr | ICMPv6 Router Advertisement |
+             +             +                               +
|             |             |                               |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+

```

### **2.2. Attack vector based on IPv6 fragmentation**

This section presents a different attack vector, which has been found to be effective against all implementations of RA-Guard. The basic idea behind this attack vector is that if the forged ICMPv6 Router Advertisement is fragmented into at least two fragments, the layer-2



device implementing "RA-Guard" would be unable to identify the attack packet, and would thus fail to block it.

A first variant of this attack vector would be an original ICMPv6 Router Advertisement message preceded with a Destination Options Header, that results in two fragments. The following figure illustrates the "original" attack packet, prior to fragmentation, and the two resulting fragments which are actually sent as part of the attack.

Original packet:

```

+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|NH=60|          |NH=58|          |          |          |
+---+---+      +---+---+          +          +
| IPv6 header |          Dst Opt Hdr          | ICMPv6 RA |
+          +          +          +
|          |          |          |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

First fragment:

```

+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|NH=44|          |NH=60|          |NH=58|          |
+---+---+      +---+---+      +---+---+          +
| IPv6 Header |   Frag Hdr   |   Dst Opt Hdr   |
+          +          +          +
|          |          |          |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Second fragment:

```

+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|NH=44|          |NH=60|          |          |          |
+---+---+      +---+---+      +          +          +
| IPv6 header |   Frag Hdr   | Dst Opt Hdr | ICMPv6 RA |
+          +          +          +          +
|          |          |          |          |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

It should be noted that the "Hdr Ext Len" field of the Destination Options Header is present in the first fragment (rather than the second). Therefore, it is impossible for a device processing only the second fragment to locate the ICMPv6 header contained in that fragment, since it is unknown how many bytes should be "skipped" to get to the next header following the Destination Options Header.





Thus, by leveraging the use of the Fragment Header together with the use of the Destination Options header, the attacker is able to conceal the type and contents of the ICMPv6 message he is sending (an ICMPv6 Router Advertisement in this example). Unless the layer-2 device were to implement IPv6 fragment reassembly, it would be impossible for the device to identify the ICMPv6 type of the message.

A layer-2 device could, however, at least detect that that an ICMPv6 message (or some type) is being sent, since the "Next Header" field of the Destination Options header contained in the first fragment is set to "58" (ICMPv6).

This idea can be taken further, such that it is also impossible for the layer-2 device to detect that the attacker is sending an ICMPv6 message in the first place. This can be achieved with an original ICMPv6 Router Advertisement message preceded with two Destination Options Headers, that results in two fragments. The following figure illustrates the "original" attack packet, prior to fragmentation, and the two resulting packets which are actually sent as part of the attack.



Original packet:

```

+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|NH=60|          |NH=60|          |NH=58|          |          |
+---+---+      +---+---+      +---+---+      +          +
| IPv6 header  | Dst Opt Hdr | Dst Opt Hdr | ICMPv6 RA |
+              +              +              +              +
|              |              |              |              |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

First fragment:

```

+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|NH=44|          |NH=60|          |NH=60|          |          |
+---+---+      +---+---+      +---+---+      +          +
| IPv6 header  | Frag Hdr  |          Dst Opt Hdr          |
+              +              +              +              +
|              |              |              |              |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Second fragment:

```

+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|NH=44|          |NH=60|          |          |NH=58|          |          |
+---+---+      +---+---+      +          +---+---+      +          +
| IPv6 header  | Frag Hdr  | Dst O Hdr | Dst Opt Hdr | ICMPv6 RA |
+              +              +              +              +
|              |              |              |              |              |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

In this variant, the "Next Header" field of the Destination Options header contained in the first fragment is set "60" (Destination Options header), and thus it is impossible for a device processing only the first fragment to detect that an ICMPv6 message is being sent in the first place.

The second fragment presents the same challenges as the second fragment of the previous variant. That is, it would be impossible for a device processing only the second fragment to locate the second Destination Options header (and hence the ICMPv6 header), since the "Hdr Ext Len" field of the first Destination Options header is present in the first fragment (rather than the second).



### 3. RA-Guard implementation advice

The following filtering rules must be implemented as part of an "RA-Guard" implementation on those ports that are not allowed to send ICMPv6 Router Advertisement messages, such that the vulnerabilities discussed in this document are eliminated:

1. If the IPv6 Source Address of the packet is not a link-local address (fe80::/10), RA-Guard must pass the packet.

RATIONALE: [Section 6.1.2 of \[RFC4861\]](#) requires nodes to discard Router Advertisement messages if their IPv6 Source Address is not a link-local address.

2. If the Hop Limit is not 255, RA-Guard must pass the packet.

RATIONALE: [Section 6.1.2 of \[RFC4861\]](#) requires nodes to discard Router Advertisement messages if their Hop Limit is not 255.

3. RA-Guard must parse the IPv6 entire header chain present in the packet, to identify whether the packet is a Router Advertisement message.

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

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

4. When parsing the IPv6 header chain, if the packet is a first-fragment (i.e., a packet containing a Fragment Header with the Fragment Offset set to 0) and it fails to contain the entire IPv6 header chain (i.e., all the headers starting from the IPv6 header up to, and including, the upper-layer header), RA-Guard must drop the packet, and should log the packet drop event in an



implementation-specific manner as a security fault.

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

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

5. When parsing the IPv6 header chain, if the packet is identified to be an ICMPv6 Router Advertisement message, RA-Guard must drop the packet, and should log the packet drop event in an implementation-specific manner as a security fault.

RATIONALE: By definition, Router Advertisement messages MUST originate on-link, MUST have a link-local IPv6 Source Address, and MUST have a Hop Limit value of 255. [[RFC4861](#)].

6. In all other cases, RA-Guard must pass the packet as usual.

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

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

In order to protect current end-node IPv6 implementations, Rule #4 has been defined as a default rule to drop packets that cannot be positively identified as not being Router Advertisement (RA) messages (because the packet is a fragment that fails to include the entire IPv6 header chain). This means that, at least in theory, RA-Guard could result in false-positive blocking of some legitimate non-RA





packets that could not be positively identified as being non-RA. In order to reduce the likelihood of false positives, Rule #1 and Rule #2 require that packets that would not pass the required validation checks for RA messages ([Section 6.1.2 of \[RFC4861\]](#)) be passed without further inspection. In any case, as noted in [\[I-D.ietf-6man-oversized-header-chain\]](#), IPv6 packets that fail to include the entire IPv6 header chain are virtually impossible to police with state-less filters and firewalls, and hence are unlikely to survive in real networks. [\[I-D.ietf-6man-oversized-header-chain\]](#) requires that hosts employing fragmentation include the entire IPv6 header chain in the first fragment (the fragment with the Fragment Offset set to 0), thus eliminating the aforementioned false positives.

This filtering policy assumes that host implementations require that the IPv6 Source Address of ICMPv6 Router Advertisement messages be a link-local address, and that they discard the packet if this check fails, as required by the current IETF specifications [\[RFC4861\]](#). Additionally, it assumes that hosts require the Hop Limit of Neighbor Discovery messages to be 255, and discard those packets otherwise.

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

Finally, we note that IPv6 implementations that allow overlapping fragments (i.e. that do not comply with [\[RFC5722\]](#)) might still be subject of RA-based attacks. However, a recent assessment of IPv6 implementations [\[SI6-FRAG\]](#) with respect to their fragment reassembly policy seems to indicate that most current implementations comply with [\[RFC5722\]](#).



#### **4. Other Implications**

A similar concept to that of "RA-Guard" has been implemented for protecting against forged DHCPv6 messages. Such protection can be circumvented with the same techniques discussed in this document, and the counter-measures for such evasion attack are analogous to those described in [Section 3](#) of this document.

[DHCPv6-Shield] specifies a mechanism to protect against rogue DHCPv6 servers, while taking into consideration the evasion techniques discussed in this document.

## **5. IANA Considerations**

This document has no actions for IANA.

## 6. Security Considerations

This document describes a number of techniques that have been found to be effective to circumvent popular RA-Guard implementations, and provides advice to RA-Guard implementations such that those evasion vulnerabilities are eliminated.

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

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

We note that most effective and efficient mitigation for these attacks would be to prohibit the use of IPv6 fragmentation with Router Advertisement messages (as proposed by [\[I-D.ietf-6man-nd-extension-headers\]](#)), such that the RA-Guard functionality is easier to implement. However, since such mitigation would require an update to existing implementations, it cannot be relied upon in the short or near term.

Finally, we note that RA-Guard only mitigates attack vectors based on ICMPv6 Router advertisement messages. Protection against similar attacks based on other messages (such as DHCPv6) is considered out of the scope of this document, and left for other documents(e.g. [\[DHCPv6-Shield\]](#)).



## **7. Acknowledgements**

The author would like to thank Ran Atkinson, who provided very detailed comments and suggested text that was incorporated into this document.

The author would like to thank Ran Atkinson, Karl Auer, Robert Downie, Washam Fan, David Farmer, Marc Heuse, Nick Hilliard, Ray Hunter, Joel Jaeggli, Simon Perreault, Arturo Servin, Gunter van de Velde, James Woodyatt, and Bjoern A. Zeeb, for providing valuable comments on earlier versions of this document.

The author would like to thank Arturo Servin, who presented this document at IETF 81.

This document resulted from the project "Security Assessment of the Internet Protocol version 6 (IPv6)" [[CPNI-IPv6](#)], carried out by Fernando Gont on behalf of the UK Centre for the Protection of National Infrastructure (CPNI). The author would like to thank the UK CPNI, for their continued support.





## **8. References**

### **8.1. Normative References**

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", [RFC 4303](#), December 2005.
- [RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", [RFC 4861](#), September 2007.
- [RFC5722] Krishnan, S., "Handling of Overlapping IPv6 Fragments", [RFC 5722](#), December 2009.
- [RFC6105] Levy-Abegnoli, E., Van de Velde, G., Popoviciu, C., and J. Mohacsi, "IPv6 Router Advertisement Guard", [RFC 6105](#), February 2011.
- [RFC6564] Krishnan, S., Woodyatt, J., Kline, E., Hoagland, J., and M. Bhatia, "A Uniform Format for IPv6 Extension Headers", [RFC 6564](#), April 2012.
- [I-D.ietf-6man-oversized-header-chain]  
Gont, F. and V. Manral, "Security and Interoperability Implications of Oversized IPv6 Header Chains", [draft-ietf-6man-oversized-header-chain-02](#) (work in progress), November 2012.
- [I-D.ietf-6man-nd-extension-headers]  
Gont, F., "Security Implications of IPv6 Fragmentation with IPv6 Neighbor Discovery", [draft-ietf-6man-nd-extension-headers-01](#) (work in progress), November 2012.

### **8.2. Informative References**

- [RFC6104] Chown, T. and S. Venaas, "Rogue IPv6 Router Advertisement Problem Statement", [RFC 6104](#), February 2011.
- [DHCPv6-Shield]  
Gont, F., "DHCPv6-Shield: Protecting Against Rogue DHCPv6 Servers", IETF Internet Draft, [draft-gont-opsec-dhcpv6-shield](#), work in progress, May 2012.



## [CPNI-IPv6]

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

[NDPMon] "NDPMon - IPv6 Neighbor Discovery Protocol Monitor",  
<<http://ndpmon.sourceforge.net/>>.

[rafixd] "rafixd", <<http://www.kame.net/dev/cvsweb2.cgi/kame/kame/kame/rafixd/>>.

[ramond] "ramond", <<http://ramond.sourceforge.net/>>.

## [SI6-FRAG]

SI6 Networks, "IPv6 NIDS evasion and improvements in IPv6 fragmentation/reassembly", 2012, <<http://blog.si6networks.com/2012/02/ipv6-nids-evasion-and-improvements-in.html>>.

## [SI6-IPv6]

"SI6 Networks' IPv6 toolkit",  
<<http://www.si6networks.com/tools/ipv6toolkit>>.

## [THC-IPv6]

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



## [Appendix A](#). Assessment tools

[SI6-IPV6] is a publicly-available set of tools (for Linux, \*BSD, and Mac OS) that implements the techniques described in this document.

[THC-IPV6] is a publicly-available set of tools (for Linux) that implements some of the techniques described in this document.

Author's Address

Fernando Gont  
Centre for the Protection of National Infrastructure

Email: [fgont@si6networks.com](mailto:fgont@si6networks.com)

URI: <http://www.cpni.gov.uk>