opsec F. Gont
Internet-Draft SI6 Networks / UTN-FRH
Intended status: Best Current Practice W. Liu
Expires: August 29, 2015 G. Van de Velde
Huawei Technologies
February 25, 2015

DHCPv6-Shield: Protecting Against Rogue DHCPv6 Servers
draft-ietf-opsec-dhcpv6-shield-06

Abstract

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

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

This document specifies DHCPv6-Shield: a mechanism for protecting hosts connected to a switched network against rogue DHCPv6 servers [RFC3315]. The basic concept behind DHCPv6-Shield is that a layer-2 device filters DHCPv6 messages meant to DHCPv6 clients (henceforth "DHCPv6-server messages"), according to a number of different criteria. The most basic filtering criterion is that DHCPv6-server messages are discarded by the layer-2 device unless they are received on a specific ports of the layer-2 device.

Before the DCHPv6-Shield device is deployed, the administrator specifies the layer-2 port(s) on which DHCPv6-server messages are to be allowed. Only those ports to which a DHCPv6 server or relay is to be connected should be specified as such. Once deployed, the DHCPv6-Shield device inspects received packets, and allows (i.e. passes) DHCPv6-server messages only if they are received on layer-2 ports that have been explicitly configured for such purpose.

DHCPv6-Shield is analogous to the RA-Guard mechanism [RFC6104] [RFC6105] [RFC7113], intended for protection against rogue Router Advertisement [RFC4861] messages.

We note that DHCPv6-Shield only mitigates only DHCPv6-based attacks against hosts. Attack vectors based on other messages meant for network configuration (such as ICMPv6 Router Advertisements) are not addressed by DHCPv6-Shield itself. In a similar vein,

DHCPv6-Shielddoes not mitigate attacks against DHCPv6 servers (e.g., Denial of Service).

2. Requirements Language

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

3. Terminology

DHCPv6-Shield:

the set of filtering rules specified in this document, meant to mitigate attacks that employ DHCPv6-server packets.

DHCPv6-Shield device:

A layer-2 device (typically a layer-2 switch) that enforces the filtering policy specified in this document.

For the purposes of this document, the terms Extension Header, Header Chain, First Fragment, and Upper-layer Header are used as specified in [RFC7112]:

IPv6 Extension Header:

Extension Headers are defined in Section 4 of [RFC2460]. As a result of [RFC7045], [IANA-PROTO] provides a list of assigned Internet Protocol Numbers and designates which of those protocol numbers also represent extension headers.

First Fragment:

An IPv6 fragment with fragment offset equal to 0.

IPv6 Header Chain:

The header chain contains an initial IPv6 header, zero or more IPv6 extension headers, and optionally, a single upper-layer header. If an upper-layer header is present, it terminates the header chain; otherwise the "No Next Header" value (Next Header = 59) terminates it.

The first member of the header chain is always an IPv6 header. For a subsequent header to qualify as a member of the header chain, it must be referenced by the "Next Header" field of the previous member of the header chain. However, if a second IPv6

header appears in the header chain, as is the case when IPv6 is tunneled over IPv6, the second IPv6 header is considered to be an upper-layer header and terminates the header chain. Likewise, if an Encapsulating Security Payload (ESP) header appears in the header chain it is considered to be an upper-layer header and it terminates the header chain.

Upper-layer Header:

In the general case, the upper-layer header is the first member of the header chain that is neither an IPv6 header nor an IPv6 extension header. However, if either an ESP header, or a second IPv6 header occur in the header chain, they are considered to be upper layer headers and they terminate the header chain.

Neither the upper-layer payload, nor any protocol data following the upper-layer payload, is considered to be part of the header chain. In a simple example, if the upper-layer header is a TCP header, the TCP payload is not part of the header chain. In a more complex example, if the upper-layer header is an ESP header, neither the payload data, nor any of the fields that follow the payload data in the ESP header are part of the header chain.

4. DHCPv6-Shield Configuration

Before being deployed for production, the DHCPv6-Shield device is explicitly configured with respect to which layer-2 ports are allowed to receive DHCPv6 packets destined to DHCPv6 clients (i.e. DHCPv6-server messages). Only those layer-2 ports explicitly configured for such purpose will be allowed to receive DHCPv6 packets to DHCPv6 clients.

5. DHCPv6-Shield Implementation Advice

The following are the filtering rules that are enforced as part of a DHCPv6-Shield implementation on those ports that are not allowed to receive DHCPv6 packets to DHCPv6 clients:

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

   RATIONALE: DHCPv6-Shield implementations MUST NOT enforce a limit on the number of bytes they can inspect (starting from the beginning of the IPv6 packet), since this could introduce false-negatives: DHCP6-server packets received on ports not allowed to receive such packets could be allowed simply

   because the DHCPv6-Shield device does not parse the entire IPv6 header chain present in the packet.

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

   RATIONALE: Packets that fail to contain the IPv6 header chain could otherwise be leveraged for circumventing DHCPv6-Shield. [RFC7112] requires that the first-fragment (i.e., the fragment with the Fragment Offset set to 0) contains the entire IPv6 header chain, and allows intermediate systems such as routers to drop those packets that fail to comply with this requirement.

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

3. DHCPv6-Shield MUST provide a configuration knob that controls whether packets with unrecognized Next Header values are dropped; this configuration knob MUST default to "drop". When parsing the IPv6 header chain, if the packet contains an unrecognized Next Header value and the configuration knob is configured to "drop", DHCPv6-Shield MUST drop the packet, and ought to log the packet drop event in an implementation-specific manner as a security alert.

   RATIONALE: An unrecognized Next Header value could possibly identify an IPv6 Extension Header, and thus be leveraged to conceal a DHCPv6-server packet (since there is no way for DHCPv6-Shield to parse past unrecognized Next Header values [I-D.gont-6man-rfc6564bis]). [RFC7045] requires that nodes be configurable with respect to whether packets with unrecognized headers are forwarded, and allows the default behavior to be that such packets be dropped.

4. In all other cases, DHCPv6-Shield MUST pass the packet as usual.

NOTE: For the purpose of enforcing the DHCPv6-Shield filtering policy, an ESP header [RFC4303] should be considered to be an

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

The above indicates that if a packet is dropped due to this filtering policy, the packet drop event be logged in an implementation-specific manner as a security fault. It is useful for the logging mechanism to include a per-port drop counter dedicated to DHCPv6-Shield packet drops.

In order to protect current end-node IPv6 implementations, Rule #2 has been defined as a default rule to drop packets that cannot be positively identified as not being DHCPv6-server packets (because the packet is a fragment that fails to include the entire IPv6 header chain). This means that, at least in theory, DHCPv6-Shield could result in false-positive blocking of some legitimate (non DHCPv6-server) packets. However, as noted in [RFC7112], 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. [RFC7112] requires that hosts employing fragmentation include the entire IPv6 header chain in the first fragment (the fragment with the Fragment Offset set to 0), thus eliminating the aforementioned false positives.

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

Finally, we note that IPv6 implementations that allow overlapping fragments (i.e. that do not comply with [RFC5722]) might still be subject of DHCPv6-based attacks. However, a recent assessment of IPv6 implementations [SI6-FRAG] with respect to their fragment reassembly policy seems to indicate that most current implementations comply with [RFC5722].

6. IANA Considerations

This document has no actions for IANA.

7. Security Considerations

The recommendations in this document represent the ideal behavior of a DHCPv6 shield device. However, in order to implement DHCPv6 shield on the fast path, it may be necessary to limit the depth into the packet that can be scanned before giving up. In circumstances where there is such a limitation, it is recommended that implementations drop packets after attempting to find a protocol header up to that limit, whatever it is. Ideally, such devices should be configurable with a list of protocol header identifiers so that if new transport protocols are standardized after the device is released, they can be added to the list of protocol header types that the device recognizes. Since any protocol header that is not a UDP header would be passed by the DHCPv6 shield algorithm, this would allow such devices to avoid blocking the use of new transport protocols. When an implementation must stop searching for recognizable header types in a packet due to such limitations, whether the device passes or drop that packet SHOULD be configurable.

The mechanism specified in this document can be used to mitigate DHCPv6-based attacks against hosts. Attack vectors based on other messages meant for network configuration (such as ICMPv6 Router Advertisements) are out of the scope of this document. Additionally, the mechanism specified in this document does not mitigate attacks against DHCPv6 servers (e.g., Denial of Service).

If deployed in layer-2 domain with several cascading switches, there will be an ingress port on the host's local switch which will need to be enabled for receiving DHCPv6-server messages. However, this local switch will be reliant on the upstream devices to have filtered out rogue DHCPv6-server messages, as the local switch has no way of determining which upstream DHCP-server messages are valid. Therefore, in order to be effective DHCPv6 Shield should be deployed and enabled on all layer-2 switches of a given layer-2 domain.

As noted in Section 5, IPv6 implementations that allow overlapping fragments (i.e. that do not comply with [RFC5722]) might still be subject of DHCPv6-based attacks. However, most current implementations seem to comply with [RFC5722], and hence forbid IPv6 overlapping fragments.

We note that if an attacker sends a fragmented DHCPv6 packet on a port not allowed to receive 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.

Additionally, we note that the security of a site employing DHCPv6 Shield could be further improved by deploying [I-D.ietf-savi-dhcp], to mitigate IPv6 address spoofing attacks.

Finally, we note that other mechanisms for mitigating attacks based on DHCPv6-server messages are available that have different deployment considerations. For example, [I-D.ietf-dhc-secure-dhcpv6] allows for authentication of DHCPv6-server packets if the IPv6 addresses of the DHCPv6 servers can be pre-configured at the client nodes.

8. Acknowledgements

The authors would like to thank (in alphabetical order) Ben Campbell, Jean-Michel Combes, Sheng Jiang, Ted Lemon, Pete Resnick, Juergen Schoenwaelder, Carsten Schmoll, Robert Sleigh, Donald Smith, Mark Smith, Hannes Tschofenig, Eric Vyncke, and Qin Wu, for providing valuable comments on earlier versions of this document.

Part of Section 3 of this document was borrowed from [RFC7112], authored by Fernando Gont, Vishwas Manral, and Ron Bonica.

This document is heavily based on the document [RFC7113] authored by Fernando Gont. Thus, the authors would like to thank Ran Atkinson, Karl Auer, Robert Downie, Washam Fan, David Farmer, Mike Heard, 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 [RFC7113], on which this document is based.

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

[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.

[RFC7112] Gont, F., Manral, V., and R. Bonica, "Implications of Oversized IPv6 Header Chains", RFC 7112, January 2014.

[RFC7045] Carpenter, B. and S. Jiang, "Transmission and Processing of IPv6 Extension Headers", RFC 7045, December 2013.

9.2. Informative References

[I-D.ietf-dhc-secure-dhcpv6] Jiang, S. and S. Shen, "Secure DHCPv6 Using CGAs", draft-ietf-dhc-secure-dhcpv6-07 (work in progress), September 2012.

[I-D.gont-6man-rfc6564bis] Gont, F., Will, W., Krishnan, S., and H. Pfeifer, "IPv6 Universal Extension Header", draft-gont-6man-rfc6564bis-00 (work in progress), April 2014.

[RFC6104] Chown, T. and S. Venaas, "Rogue IPv6 Router Advertisement Problem Statement", RFC 6104, February 2011.

[RFC6105] Levy-Abegnoli, E., Van de Velde, G., Popoviciu, C., and J. Mohacsi, "IPv6 Router Advertisement Guard", RFC 6105, February 2011.

[RFC7113] Gont, F., "Implementation Advice for IPv6 Router Advertisement Guard (RA-Guard)", RFC 7113, February 2014.

[IANA-PROTO] Internet Assigned Numbers Authority, "Protocol Numbers", February 2013, <http://www.iana.org/assignments/protocol-numbers/protocol-numbers.txt>.

[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>.

[I-D.ietf-savi-dhcp]
Bi, J., Wu, J., Yao, G., and F. Baker, "SAVI Solution for DHCP", draft-ietf-savi-dhcp-34 (work in progress), February 2015.

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

Authors' Addresses

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

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

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

Email: liushucheng@huawei.com

Gunter Van de Velde

Huawei Technologies

Email: gunterv.velde@huawei.com