Network Working Group
Internet-Draft
Updates: 4862 (if approved)
Intended status: Standards Track
Expires: August 9, 2013

R. Asati
H. Singh
W. Beebee
Cisco Systems, Inc.
E. Dart
Lawrence Berkeley National Laboratory
W. George
Time Warner Cable
C. Pignataro
Cisco Systems, Inc.

February 5, 2013

Enhanced Duplicate Address Detection
draft-ietf-6man-enhanced-dad-02

Abstract

Appendix A of the IPv6 Duplicate Address Detection (DAD) document in RFC 4862 discusses Loopback Suppression and DAD. However, RFC 4862 does not settle on one specific automated means to detect loopback of Neighbor Discovery (ND of RFC 4861) messages used by DAD. Several service provider communities have expressed a need for automated detection of looped backed ND messages used by DAD. This document includes mitigation techniques and then outlines the Enhanced DAD algorithm to automate detection of looped back IPv6 ND messages used by DAD. For network loopback tests, the Enhanced DAD algorithm allows IPv6 to self-heal after a loopback is placed and removed. Further, for certain access networks the document automates resolving a specific duplicate address conflict.

Status of this Memo

Copyright Notice

Table of Contents

1. Terminology

- DAD-failed state - Duplication Address Detection failure as specified in [RFC4862]. Failure also includes if the Target Address is optimistic. Optimistic DAD is specified in [RFC4429].

- Looped back message - also referred to as a reflected message. The message sent by the sender is received by the sender due to the network or a Upper Layer Protocol on the sender looping the message back.

- Loopback - A function in which the router's interface (or the circuit to which the router's interface is connected) is looped back or connected to itself. Loopback causes packets sent by the interface to be received by the interface, and results in interface unavailability for regular data traffic forwarding. See more details in section 9.1 of [RFC1247]. The Loopback function is commonly used in an interface context to gain information on the quality of the interface, by employing mechanisms such as ICMPv6 pings, bit-error tests, etc. In a circuit context, it is used in wide area environments including optical dense wave division multiplexing (DWDM) and SONET/SDH for fault isolation (e.g. by placing a loopback at different geographic locations along the path of a wide area circuit to help locate a circuit fault). The Loopback function may be employed locally or remotely.

- NS(DAD) - shorthand notation to denote an NS with unspecified IPv6 source-address issued during DAD.

2. Introduction

Appendix A of [RFC4862] discusses Loopback Suppression and Duplicate Address Detection (DAD). However, [RFC4862] does not settle on one specific automated means to detect loopback of ND messages used by DAD. One specific DAD message is a Neighbor Solicitation (NS), specified in [RFC4861]. The NS is issued by the network interface of an IPv6 node for DAD. Another message involved in DAD is a Neighbor Advertisement (NA). The Enhanced DAD algorithm proposed in this document focuses on detecting an NS looped back to the transmitting interface during the DAD operation. Detecting a looped back NA is of no use because no problems with DAD will occur if a node receives a looped back NA. Detection of any other looped back ND messages outside of the DAD operation is not critical and thus this document does not cover such detection. The document also includes a Mitigation section that discusses means already available to mitigate the loopback problem.

Recently, service providers have reported a problem with DAD that is caused by looped back NS messages. The following is a description of the circumstances under which the problem arises. Loopback testing for troubleshooting purposes is underway on a circuit connected to an interface on a router. The interface on the router is enabled for IPv6. The interface issues a NS for the IPv6 link-local address DAD. The NS is reflected back to the router interface due to the loopback condition of the circuit, and the router interface enters a DAD-failed state. After the circuit troubleshooting has concluded and the loopback condition is removed, IPv4 will return to operation without further manual intervention. However, IPv6 will remain in DAD-failed state until manual intervention on the router restores IPv6 to operation.

There are other conditions which will also trigger similar problems with DAD Loopback. While the following example is not a common configuration, it has occurred in a large service provider network. It is necessary to address it in the proposed solution because the trigger scenario has the potential to cause significant IPv6 service outages when it does occur. Two broadband modems in the same location are served by the same service provider and both modems are served by one access concentrator and one layer 3 interface on the access concentrator. The two modems have the Ethernet ports of each modem connected to a network hub. The access concentrator serving the modems is the first-hop IPv6 router for the modems. The access concentrator also supports proxying of DAD messages. Each modem is enabled for at least data services. The network interface of the access concentrator serving the two broadband modems is enabled for IPv6 and the interface issues a NS(DAD) message for the IPv6 link-local address. The NS message reaches one modem first and this modem sends the message to the network hub which sends the message to the second modem which forwards the message back to the access concentrator. The looped back NS message causes the network interface on the access concentrator to be in a DAD-failed state. Such a network interface typically serves up to 100 thousand broadband modems causing all the modems (and hosts behind the modems) to fail to get IPv6 online on the access network. Additionally, it may be tedious for the access concentrator to find out which of the six thousand or more homes looped back the DAD message. Clearly there is a need for automated detection of looped back NS messages during DAD operations by a node.

3. Operational Mitigation Options

Two mitigation options are described below. The mechanisms do not require any change to existing implementations.

3.1. Disable DAD on Interface

One can disable DAD on an interface and then there is no NS(DAD) issued to be looped back. DAD is disabled by setting the interface's DupAddrDetectTransmits variable to zero. While this mitigation may be the simplest the mitigation has three drawbacks.

It would likely require careful analysis of configuration on such point-to-point interfaces, a one-time manual configuration on each of such interfaces, and more importantly, genuine duplicates in the link will not be detected.

A Service Provider router such as an access concentrator or network core router SHOULD support this mitigation strategy.

3.2. Dynamic Disable/Enable of DAD Using Layer 2 Protocol

It is possible that one or more layer 2 protocols include provisions to detect the existence of a loopback on an interface circuit, usually by comparing protocol data sent and received. For example, PPP uses magic number (section 6.4 of [RFC1661]) to detect a loopback on an interface.

When a layer 2 protocol detects that a loopback is present on an interface circuit, the device MUST temporarily disable DAD on the interface, and when the protocol detects that a loopback is no longer present (or the interface state has changed), the device MUST (re-)enable DAD on that interface.

This solution requires no protocol changes. This solution SHOULD be enabled by default, and MUST be a configurable option.

This mitigation has several benefits. They are

1. It leverages layer 2 protocol's built-in loopback detection capability, if available.

2. It scales better since it relies on an event-driven model which requires no additional state or timer. This may be a significant scaling consideration on devices with hundreds or thousands of interfaces that may be in loopback for long periods of time (such as while awaiting turn-up or during long-duration intrusive bit error rate tests).

3.3. Operational Considerations

The mitigation options discussed in the document do not require the devices on both ends of the circuit to support the mitigation functionality simultaneously, and do not propose any capability negotiation. The mitigation options discussed in this document are effective for unidirectional circuit or interface loopback (i.e. the the loopback is placed in one direction on the circuit, rendering the other direction non-operational).

The mitigation options may not be effective for the bidirectional loopback (i.e. the loopback is placed in both directions of the circuit interface, so as to identify the faulty segment) if only one device followed a mitigation option specified in this document, since the other device would follow current behavior and disable IPv6 on that interface due to DAD until manual intervention restores it.

This is nothing different from what happens today (without the solutions proposed by this document) in case of unidirectional loopback. Hence, it is expected that an operator would resort to manual intervention for the devices not compliant with this document, as usual.

4. The Enhanced DAD Algorithm

The Enhanced DAD algorithm covers detection of a looped back NS(DAD) message. The document proposes use of the Nonce Option specified in the SEND document of [RFC3971]. The nonce is a random number as specified in [RFC3971]. If SEND is enabled on the router and the router also supports the Enhanced DAD algorithm (specified in this document), there is integration with the Enhanced DAD algorithm and SEND. See more details in the Impact on SEND section. Since a nonce is used only once, DAD for each IPv6 address of an interface uses a different nonce.

When the IPv6 network interface issues a NS(DAD) message, the interface includes the Nonce Option in the NS(DAD) message and saves the nonce in local store. Subsequently if the interface receives an identical NS(DAD) message, the interface logs a system management message, updates any statistics counter, and drops the looped back NS(DAD). Additionally the interface continues to issue subsequent probes until a termination condition (to be defined) is detected. The DupAddrDetectTransmits value is ignored by the algorithm on detection of a looped back NS(DAD). Note [RFC4861] already randomizes issuing the first probe and issues up to three probes; note [I-D.ietf-6man-impatient-nud] has intentions to change probe behavior of [RFC4861]. Additionally, certain networks take a few

minutes to loop back a packet to a sender. Hence a variable is needed to help the interface decide how long to wait for the DAD process to complete. For example, [RFC4862] waits for two seconds to see if any message is received to signal a DAD failure, and if not, DAD is completed. This document defines a new variable in WAIT_TIME_LOOPBACK. For example, WAIT_TIME_LOOPBACK is two seconds for [RFC4862] behavior and five minutes for another network. The interface also uses an exponentially backoff algorithm (pick one of several available) to issue probes.

If the interface receives a NS(DAD) message with a different nonce but TargetAddress matches a tentative or optimistic address on the interface, the interface logs a DAD-failed system management message, updates any statistics, and behaves identical to the behavior specified in [RFC4862] for DAD failure.

Six bytes of random nonce is sufficiently large for nonce collisions. However if there is a collision because two nodes generated the same random nonce (that are using the same Target address in their NS(DAD)), then the algorithm will incorrectly detect a looped back NS(DAD) when the NS(DAD) was issued to signal a genuine duplicate. Since each looped back NS(DAD) event is logged to system management, the administrator of the network will have to intervene manually.

The algorithm is capable of detecting any ND solicitation (NS and Router Solicitation) or advertisement (NA and Router Advertisement) that is looped back. However, saving a nonce and nonce related data for all ND messages has impact on memory of the node and also adds the algorithm state to a substantially larger number of ND messages. Therefore this document does not recommend using the algorithm outside of the DAD operation by an interface on a node.

4.1. General Rules

If an IPv6 node implements the Enhanced DAD algorithm, the node MUST implement detection of looped back NS(DAD) messages during DAD for an interface address.

4.2. Processing Rules for Senders

If a node has been configured to use the Enhanced DAD algorithm, when sending a NS(DAD) for a tentative or optimistic interface address the sender MUST generate a random nonce associated with the interface address, MUST save the nonce, and MUST include the nonce in the Nonce Option included in the NS(DAD). If a looped back NS(DAD) is detected by the interface, the interface ignores the DupAddrDetectTransmits and issues subsequent probes forever in an exponential backoff transmission until a termination condition is detected.

4.3. Processing Rules for Receivers

If the node has been configured to use the Enhanced DAD algorithm and an interface on the node receives any NS(DAD) message where the target address matches the interface address (in tentative or optimistic state), the receiver compares the nonce, if any, is included in the message with any saved nonce on the receiving interface. If a match is found, the node SHOULD log a system management message, SHOULD update any statistics counter, and MUST drop the received message. If the received NS(DAD) message includes a nonce and no match is found with any saved nonce, the node SHOULD log a system management message for DAD-failed and SHOULD update any statistics counter. TODO: Define termination condition.

4.4. Impact on SEND

The SEND document uses the Nonce Option in the context of matching an NA with an NS. However, no text in SEND has an explicit mention of detecting looped back ND messages. If this document updates [RFC4862], SEND should be updated to integrate with the Enhanced DAD algorithm. A minor update to SEND would be to explicitly mention that the nonce in SEND is also used by SEND to detect looped back NS messages during DAD operations by the node. In a mixed SEND environment with SEND and unsecured nodes, the lengths of the nonce used by SEND and unsecured nodes MUST be identical.

4.5. Changes to RFC 4862

The following text is added to [RFC4862].

A network interface of an IPv6 node SHOULD implement the Enhanced DAD algorithm. For example, if the interface on an IPv6 node is connected to a circuit that supports loopback testing, then the node should implement the Enhanced DAD algorithm that allows the IPv6 interface to self-heal after loopback testing is ended on the circuit. Another example is when the IPv6 interface resides on an access concentrator running DAD Proxy. The interface supports up to 100 thousand IPv6 clients (broadband modems) connected to the interface. If the interface performs DAD for its IPv6 link-local address and if the DAD probe is reflected back to the interface, the interface is stuck in DAD failed state and IPv6 services to the 100 thousand clients is denied. Disabling DAD for such an IPv6 interface on an access concentrator is not an option because the network also needs to detect genuine duplicates in the interface downstream network. The Enhanced DAD algorithm also facilitates detecting a genuine duplicate for the interface on the access concentrator. See the Actions to Perform on Detecting a Genuine Duplicate section of the Enhanced DAD document.

5. Actions to Perform on Detecting a Genuine Duplicate

As described in paragraphs above the nonce can also serve to detect genuine duplicates even when the network has potential for looping back ND messages. When a genuine duplicate is detected, the node follows the manual intervention specified in section 5.4.5 of [RFC4862]. However, in certain networks such as an access network if the genuine duplicate matches the tentative or optimistic IPv6 address of a network interface of the access concentrator, automated actions are proposed.

One access network is a cable broadband deployment where the access concentrator is the first-hop IPv6 router to several thousand broadband modems. The router also supports proxying of DAD messages. The network interface on the access concentrator initiates DAD for an IPv6 address and detects a genuine duplicate due to receiving an NS(DAD) or an NA message. On detecting such a duplicate the access concentrator logs a system management message, drops the received ND message, and blocks the modem on whose layer 2 service identifier the NS(DAD) or NA message was received on.

The network described above follows a trust model where a trusted router serves un-trusted IPv6 host nodes. Operators of such networks have a desire to take automated action if a network interface of the trusted router has a tentative or optimistic address duplicate with a host served by trusted router interface. Any other network that follows the same trust model MAY use the automated actions proposed in this section.

6. Security Considerations

The nonce can be exploited by a rogue deliberately changing the nonce to fail the looped back detection specified by the Enhanced DAD algorithm. SEND is recommended for this exploit. For any mitigation suggested in the document such as disabling DAD has an obvious security issue before a remote node on the link can issue reflected NS(DAD) messages. Again, SEND is recommended for this exploit.

7. IANA Considerations

None.

8. Acknowledgements

Thanks (in alphabetical order by first name) to Dmitry Anipko, Eric

Levy-Abegnoli, Erik Nordmark, Fred Templin, Michael Sinatra, Ray Hunter, Suresh Krishnan, and Tassos Chatzithomaoglou for their guidance and review of the document. Thanks to Thomas Narten for encouraging this work. Thanks to Steinar Haug and Scott Beuker for describing the use cases.

9. Normative References

[I-D.ietf-6man-impatient-nud]
Nordmark, E. and I. Gashinsky, "Neighbor Unreachability Detection is too impatient", draft-ietf-6man-impatient-nud-05 (work in progress), October 2012.

[RFC1247] Moy, J., "OSPF Version 2", RFC 1247, July 1991.

[RFC1661] Simpson, W., "The Point-to-Point Protocol (PPP)", STD 51, RFC 1661, July 1994.

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

[RFC3971] Arkko, J., Kempf, J., Zill, B., and P. Nikander, "SEcure Neighbor Discovery (SEND)", RFC 3971, March 2005.

[RFC4429] Moore, N., "Optimistic Duplicate Address Detection (DAD) for IPv6", RFC 4429, April 2006.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, September 2007.

[RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.

Appendix A. Changes from the -01 version

1. Changed the algorithm in section 4 and other sections to not suppress subsequent probes and instead probe forever subject to a termination condition.

2. Minor edits were made for more clarified text.

Authors' Addresses

Rajiv Asati
Cisco Systems, Inc.
7025 Kit Creek road
Research Triangle Park, NC 27709-4987
USA

Email: rajiva@cisco.com
URI: http://www.cisco.com/

Hemant Singh
Cisco Systems, Inc.
1414 Massachusetts Ave.
Boxborough, MA 01719
USA

Phone: +1 978 936 1622
Email: shemant@cisco.com
URI: http://www.cisco.com/

Wes Beebee
Cisco Systems, Inc.
1414 Massachusetts Ave.
Boxborough, MA 01719
USA

Phone: +1 978 936 2030
Email: wbeebee@cisco.com
URI: http://www.cisco.com/

Eli Dart
Lawrence Berkeley National Laboratory
1 Cyclotron Road, Berkeley, CA 94720
USA

Email: dart@es.net
URI: http://www.es.net/

Wes George
Time Warner Cable
13820 Sunrise Valley Drive
Herndon, VA 20171
USA

Email: wesley.george@twcable.com

Carlos Pignataro
Cisco Systems, Inc.
7200-12 Kit Creek Road
Research Triangle Park, NC 27709
USA

Email: cpignata@cisco.com
URI: http://www.cisco.com/