Network Working Group
Internet-Draft
Intended status: Informational
Expires: September 1, 2015

A. Yourtchenko
cisco
E. Nordmark
Arista Networks
February 28, 2015

A survey of issues related to IPv6 Duplicate Address Detection
draft-yourtchenko-6man-dad-issues-01

Abstract

This document enumerates the practical issues observed with respect to Duplicate Address Detection.

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 September 1, 2015.

Table of Contents

1. Introduction 3
2. Open Issues 3
2.1. Robustness: Interaction with delay in forwarding 3
2.2. Robustness: Behavior on links with unreliable multicast . 4
2.3. Robustness: Partition-join tolerance 4
2.4. Robustness: Behavior on collision 4
2.5. Energy Efficiency 5
2.6. Wake-up and L2 events 5
3. Solved Issues 5
3.1. Interaction with looped interfaces 5
3.2. Delays before an address can be used 6
4. Observations 6
4.1. Duplicate L2 address detection 6
4.2. Usage of DAD to create state 6
4.3. No support of multi-link subnets 7
4.4. Anycast Addresses and Duplicate Address Detection 7
4.5. Implementations doing DAD once per IID 7
4.6. Backwards compatibility and presence of the DAD proxies . 8
5. Acknowledgements 8
6. IANA Considerations 8
7. Security Considerations 8
8. References 8
8.1. Normative References 8
8.2. Informative References 9
Authors' Addresses 10

1. Introduction

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 [RFC2119].

Duplicate Address Detection (DAD) is a procedure in IPv6 performed on an address before it can be assigned to an interface [RFC2462]. By default it consists of sending a single multicast Neighbor Soliciation message and waiting for a response for one second. If no response is received, the address is declared to not be a duplicate. Once the address has been tested once, there is no further attempts to check for duplicates (unless the interface is re-initialized).

On one hand, it is mandatory for all addresses. On the other hand, it is a "best effort" activity. These somewhat counter-intuitive properties result in some issues that arise related to DAD. They are listed below. The issues have been grouped to facilitate discussing them.

2. Open Issues

Whether it is due to the assumptions made in 1995, or changes in how networks are built or deployed, there are many reasons why DAD would fail to detect a duplicate even when one exists. From a historical perspective it is important to keep in mind that when DAD was designed we had two forms of IPv6 addresses; those derived from EUI-64 and statically assigned. Since the IETF has developed additional methods for address assignment like DHCPv6 and addresses that improve privacy by reducing linkability.

2.1. Robustness: Interaction with delay in forwarding

The DAD makes an assumption that if a link layer is up, the traffic can be immediately forwarded, which is frequently not the case in modern networks. Two prominent cases include the switches running Spanning Tree Protocol (STP), and bridging modems.

When a port on an STP-enabled switch comes up, it goes through three phases of Listening then Learning then Forwarding. The default is to keep it for 15 seconds in Listening and 15 seconds in Learning states. During this time no user traffic is forwarded by the switch from and to this port. Therefore, if a DAD process happens during this period it is guaranteed to not detect any duplicates. This results in DAD being ineffective for link-local and otherwise pre configured addresses.

Similarly, a modem-like device whose line status is invisible to IP stack either within the modem or to a host connected on the Ethernet side, also renders the DAD ineffective - the delay before the connectivity is established can be much longer than any DAD wait.

Some of the link types, notably cable modems, have link-specific standards to address this issue by requiring a new DAD each time the RF-side interface bounces, as well as bouncing the LAN interface triggered by the bounce of the RF interface.

Note that [I-D.ietf-6man-resilient-rs] makes the router solicitation resilent to the above cases, but there is no counterpart to make DAD robust.

2.2. Robustness: Behavior on links with unreliable multicast

DAD requires two multicast messages to pass through - the NS and NA. Thus it shows a noticeable failure rate on links that do not pass multicast reliably e.g. the 802.11a/b/g/n series of technologies. See [I-D.vyncke-6man-mcast-not-efficient] for more information.

The author's ad-hoc experimentation at IETF90 revealed the success rate of detecting the duplicate address on the IETF WiFi network being about 4 in 5. This may violate the assumptions that other protocols make.

2.3. Robustness: Partition-join tolerance

[RFC4862] explicitly mentions this problem: "Note that the method for detecting duplicates is not completely reliable, and it is possible that duplicate addresses will still exist (e.g., if the link was partitioned while Duplicate Address Detection was performed)."

In contrast, IPv4 stacks typically implement the Address Conflict Detection (ACD) from [RFC5227]. This disparity results in a less robust operation of IPv6 compared to IPv4 and is undesirable.

Note that solutions along the lines of ACD, while improving robustness, might result in more resource usage in on the links and nodes by multicasting more ND packets.

2.4. Robustness: Behavior on collision

[RFC4862] in its section "5.4.5. When Duplicate Address Detection Fails" is much more prescriptive than [RFC2462] that it superceeds. However, it has been observed that some implementations may simply reset the network interface and attempt the DAD process again. This behavior, while being more resilient in case the DAD failure is

happening erroneously, is different from what is recommended in the standard.

TBD: Do the other RFCs for address allocation require some retry behavior?

2.5. Energy Efficiency

The use of multicast messages for DAD results in some inefficiencies for both the network, in particular when multicast uses more layer 2 resources than unicast, and also has efficiency implications for hosts. Potential techniques for making DAD reliably detect and recover from duplicates might result in reduced efficiency. The impact for WiFi is shown in [I-D.desmouceaux-ipv6-mcast-wifi-power-usage].

If a node wants to "defend" its address using DAD, it has to be awake and listening on the solicited node multicast address in order to receive the DAD NS. In the low-power environments this may significantly impact the battery life of the devices.

2.6. Wake-up and L2 events

In mobile environments, node may roam in different parts of the network and also take "naps". The specification in [RFC4862] does not explicitly discuss this scenario, nor does DNA [RFC6059], so there is a room for ambiguity in implementation. This may either result in less robust DAD coverage (if the node does not perform the DAD again when an L2 event happens), or an excessive amount of multicast packets (when a node performs the dad every time L2 event happens and there is a lot of them moving within a segment).

Thus this item could be categorized as being either in the robustness or efficiency group of items.

3. Solved Issues

Some issues have been or are in the process of being solved.

3.1. Interaction with looped interfaces

[RFC4862] explicitly defines that the case of a physically looped back interface is not a failure: "If the solicitation is from the node itself (because the node loops back multicast packets), the solicitation does not indicate the presence of a duplicate address."

However, the practical experiences show that the measures described

in [RFC4862] are either incomplete or incorrectly implemented: a loopback on the interface causes DAD failure.

[I-D.ietf-6man-enhanced-dad] provides the solution to this issue.

3.2. Delays before an address can be used

Section "5.4. Duplicate Address Detection" of [RFC4862] specifies that until the DAD procedure completes, the address remains in Tentative state. In this state, any traffic to this address other than that related to DAD-related is dropped. This introduces delay between the interface getting connected to the network and an address on this interface becoming usable. For fast-moving nodes it may be a problem.

[RFC4429] introduces "Optimistic DAD" process, which addresses this. That document has some notes about potentially causing TCP RST when there is a duplicate, which can reset an existing TCP connection for the existing user of the IPv6 address. That has some overall impact on the robustness of the network and implicitly assumes that all application protocols will always retry in order to handle such an event.

4. Observations

Some issues we can't do much about in that they are more observations of what can be done.

4.1. Duplicate L2 address detection

DAD does not detect duplicate L2 addresses in all cases. Depending on the medium, it may be impossible to detect a duplicate L2 address - e.g. if this address itself is used as a determinant in order to establish the L2 connection.

4.2. Usage of DAD to create state

[RFC4862] in section "5.4. Duplicate Address Detection" states that DAD must be performed on all addresses. Given the potentially decentralized nature of address assignment in IPv6, this property is being used to prebuild the state in the network about the host's addresses - e.g. for "First Come First Served" security as described in section "3.2.3. Processing of Local Traffic" of [RFC6620].

If the delivery of the DAD_NS packets is unreliable or there are nodes on the segment which use the Optimistic DAD mechanism, state created purely on DAD_NS packets might be also unreliable. The

specific case of [RFC6620] solves the issue by triggering the recreation of state based on data packets as well, however it might not be possible in some scenarios.

4.3. No support of multi-link subnets

DAD doesn't support multi-link subnets: a multicast DAD_NS sent on one link will not be seen on the other.

[RFC6275] specifically provides one way to construct a multi-link subnet (consisting of a broadcast link and a collection of point to point tunnels). It explicitly defines the procedures for making DAD work in that topology.

[RFC4903] discusses the issues related to multi-link subnets - and given the multi-link subnets might be created in many ways, it might be prudent to keep enhancements to DAD whose sole purpose is related to multi-link subnets, to be out of scope.

One may also argue that since [RFC4861] defers the clarifications on IPv6 operation on NBMA networks to [RFC2491], it is unreasonable to expect [RFC4862] describe the operation of DAD on NBMA type links, and it is up to a link-specific document to describe such operation. (An example is cable industry, where the cable standards define it).

However, it is then unclear where to address the frequently used scenario of WiFi with blocked direct communication between the stations - whether it is supposed to be an IEEE document or IETF document ? And is there enough fundamental differences between the different NBMA models to warrant the link-specific approaches to DAD ?

4.4. Anycast Addresses and Duplicate Address Detection

Section 5.4 "Duplicate Address Detection" of [RFC4862] specifies that Duplicate Address Detection MUST NOT be performed on anycast addresses. This, stems from the fact that the anycast addresses are syntactically indistinguishable from unicast addresses. One can argue that this allows for misconfiguration if an address deemed to be anycast already exist on the network.

4.5. Implementations doing DAD once per IID

Section 5.4 of [RFC4862] mentions the implementations performing a single DAD per interface identifier, and discourages that "optimization". As the practice is emerging in the industry is to move away from the fixed interface identifiers anyhow, the necessity to perform a DAD on a per-address basis might be useful to elevate to

a requirement status.

4.6. Backwards compatibility and presence of the DAD proxies

While not being an issue as such, this is a reminder that the operation of DAD has to remain backwards compatible, both to remain cooperative with the existing hosts, and the potentially present DAD proxies as described in [RFC6957].

There are also various forms of sleep proxies [ECMA-393] [http://en.wikipedia.org/wiki/Bonjour_Sleep_Proxy] which perform handoffs of Neighbor Discovery protocol processing that need to be considered.

5. Acknowledgements

Thanks to Ole Troan for creating and curating the original list. Thanks a lot to Lorenzo Colitti, Suresh Krishnan, Hemant Singh, Hesham Soliman, Eric Vyncke, and James Woodyatt for the reviews and useful suggestions.

6. IANA Considerations

None.

7. Security Considerations

There are no additional security considerations as this document only outlines the issues observed with the current Duplicate Address Detection protocol.

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.

[RFC2462] Thomson, S. and T. Narten, "IPv6 Stateless Address Autoconfiguration", RFC 2462, December 1998.

[RFC2491] Armitage, G., Schulter, P., Jork, M., and G. Harter, "IPv6 over Non-Broadcast Multiple Access (NBMA) networks", RFC 2491, January 1999.

[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.

[RFC4903] Thaler, D., "Multi-Link Subnet Issues", RFC 4903, June 2007.

[RFC5227] Cheshire, S., "IPv4 Address Conflict Detection", RFC 5227, July 2008.

[RFC6059] Krishnan, S. and G. Daley, "Simple Procedures for Detecting Network Attachment in IPv6", RFC 6059, November 2010.

[RFC6275] Perkins, C., Johnson, D., and J. Arkko, "Mobility Support in IPv6", RFC 6275, July 2011.

[RFC6620] Nordmark, E., Bagnulo, M., and E. Levy-Abegnoli, "FCFS SAVI: First-Come, First-Served Source Address Validation Improvement for Locally Assigned IPv6 Addresses", RFC 6620, May 2012.

[RFC6957] Costa, F., Combes, J-M., Pougnard, X., and H. Li, "Duplicate Address Detection Proxy", RFC 6957, June 2013.

8.2. Informative References

[I-D.desmouceaux-ipv6-mcast-wifi-power-usage]
Desmouceaux, Y., "Power consumption due to IPv6 multicast on WiFi devices", draft-desmouceaux-ipv6-mcast-wifi-power-usage-01 (work in progress), August 2014.

[I-D.ietf-6man-enhanced-dad]
Asati, R., Singh, H., Beebee, W., Pignataro, C., Dart, E., and W. George, "Enhanced Duplicate Address Detection", draft-ietf-6man-enhanced-dad-13 (work in progress), February 2015.

[I-D.ietf-6man-resilient-rs]
Krishnan, S., Anipko, D., and D. Thaler, "Packet loss resiliency for Router Solicitations",

draft-ietf-6man-resilient-rs-04 (work in progress), October 2014.

[I-D.vyncke-6man-mcast-not-efficient]
Vyncke, E., Thubert, P., Levy-Abegnoli, E., and A. Yourtchenko, "Why Network-Layer Multicast is Not Always Efficient At Datalink Layer", draft-vyncke-6man-mcast-not-efficient-01 (work in progress), February 2014.

Authors' Addresses

Andrew Yourtchenko
cisco
6b de Kleetlaan
Diegem 1831
Belgium

Email: ayourtch@cisco.com

Erik Nordmark
Arista Networks
Santa Clara, CA
USA

Email: nordmark@arista.com