INTERNET-DRAFT   Cameron Byrne
Intended Status: Informational   T-Mobile US
Expires: July 1, 2014   December 28, 2013

IPv6 Transitional Technology IPv4 Prefix
draft-byrne-v6ops-clatip-01

Abstract

DS-Lite [RFC6333] directs IANA to reserve 192.0.0.0/29 for the B4 element. This memo generalizes that reservation to include other cases where a non-routed IPv4 interface must be numbered in an IPv6 transition solution.

Status of this Memo

Copyright and License Notice

Table of Contents

1 Introduction

DS-Lite [RFC6333] directs IANA to reserve 192.0.0.0/29 for the B4 element. This memo generalizes that IANA reservation to include other cases where a non-routed IPv4 interface must be numbered in an IPv6 transition solutions. IANA shall list 192.0.0.0/29 to be reserved for IPv6 Transitional Technology IPv4 Prefix. The result is that 192.0.0.0/29 may be used in any system that requires IPv4 addresses for backward compatibility with IPv4 communications, but does not emit IPv4 packets "on the wire".

2 The Case of 464XLAT

464XLAT [RFC6877] describes an architecture for providing IPv4 communication over an IPv6-only access network. One of the methods described in [RFC6877] is for the client side translator (CLAT) to be embedded in the host, such as a smartphone. In this scenario, the host must have an IPv4 address configured to present to the network stack and for applications to bind sockets.

3. Choosing 192.0.0.0/29

To avoid conflicts with any other network that may communicate with the CLAT, a locally unique address must be assigned.

IANA has defined a well-known range, 192.0.0.0/29, in [RFC6333], which is dedicated for DS-lite. As defined in [RFC6333], this subnet is only present between the B4 and the AFTR and never emits packets from this prefix "on the wire". 464XLAT has the same need for a non-routed IPv4 prefix. It is most prudent and effective to generalize 192.0.0.0/29 for the use of supporting IPv4 interfaces in IPv6 transition technologies rather than reserving a prefix for every possible solution.

4 Security Considerations

No new security considerations beyond what is described [RFC6333] and [RFC6877].

5 IANA Considerations

IANA is directed to generalize the reservation of 192.0.0.0/29 from DS-lite to "IPv6 Transitional Technology IPv4 Prefix".

6 References

6.1 Normative References

[RFC6333] Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", RFC6333, August 2011.

[RFC6877] Mawatari, M., Kawashima, M., and C. Byrne, "464XLAT: Combination of Stateful and Stateless Translation", RFC6877, April 2013.

Authors' Addresses

Cameron Byrne
Bellevue, WA, USA
Email: Cameron.Byrne@T-Mobile.com

IPv6 Operations
Internet-Draft
Intended status: Informational
Expires: June 8, 2014

M. Gysi
Swisscom
G. Leclanche
Viagenie
E. Vyncke, Ed.
Cisco Systems
R. Anfinsen
Altibox

December 05, 2013

Balanced Security for IPv6 Residential CPE
draft-ietf-v6ops-balanced-ipv6-security-01

Abstract

This document describes how an IPv6 residential Customer Premise Equipment (CPE) can have a balanced security policy that allows for a mostly end-to-end connectivity while keeping the major threats outside of the home. It is documenting an existing IPv6 deployment by Swisscom and allows all packets inbound/outbound EXCEPT for some layer-4 ports where attacks and vulnerabilities (such as weak passwords) are well-known. The policy is a proposed set of rules that can be used as a default setting. The set of blocked inbound and outbound ports is expected to be updated as threats come and go.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Internet access in residential IPv4 deployments generally consists of a single IPv4 address provided by the service provider for each home. The residential CPE then translates the single address into multiple private IPv4 addresses allowing more than one device in the home, but at the cost of losing end-to-end reachability. IPv6 allows all devices to have a globally unique IP address, restoring end-to-end reachability directly between any device. Such reachability is very powerful for ubiquitous global connectivity, and is often heralded as one of the significant advantages to IPv6 over IPv4. Despite this, concern about exposure to inbound packets from the IPv6 Internet (which would otherwise be dropped by the address translation function if they had been sent from the IPv4 Internet) remain.

This difference in residential default internet protection between IPv4 and IPv6 is a major concern to a sizable number of ISPs and the security policy described in this document addresses this concern without damaging IPv6 end-to-end connectivity.

The security model provided in this document is meant to be used as a pre-registered setting and potentially default one for IPv6 security in CPEs. The model departs from the "simple security" model described in [RFC6092] . It allows most traffic, including incoming unsolicited packets and connections, to traverse the CPE unless the CPE identifies the traffic as potentially harmful based on a set of rules. This policy has been deployed as a default setting in Switzerland by Swisscom for residential CPEs.

This document can be applicable to off-the-shelves CPE as well as to managed Service Provider CPE or for mobile Service Providers (where it can be centrally implemented).

2. Threats

For a typical residential network connected to the Internet over a broadband or mobile connection, the threats can be classified into:

- denial of service by packet flooding: overwhelming either the access bandwidth or the bandwidth of a slower link in the residential network (like a slow home automation network) or the CPU power of a slow IPv6 host (like networked thermostat or any other sensor type nodes);

- denial of service by Neighbor Discovery cache exhaustion [RFC6583]: the outside attacker floods the inside prefix(es) with packets with a random destination address forcing the CPE to exhaust its memory and its CPU in useless Neighbor Solicitations;

- denial of service by service requests: like sending print jobs from the Internet to an ink jet printer until the ink cartridge is empty or like filing some file server with junk data;

- unauthorized use of services: like accessing a webcam or a file server which are open to anonymous access within the residential network but should not be accessed from outside of the home network or accessing to remote desktop or SSH with weak password protection;

- exploiting a vulnerability in the host in order to get access to data or to execute some arbitrary code in the attacked host;

- trojanized host (belonging to a Botnet) can communicate via a covert channel to its master and launch attacks to Internet targets.

3. Overview

The basic goal is to provide a pre-defined security policy which aims to block known harmful traffic and allow the rest, restoring as much of end-to-end communication as possible. This pre-defined policy should be centrally updated, as threats are changing over time. It could also be a member of a list of pre-defined security policies available to an end-customer, for example together with "simple security" from [RFC6092] and a "strict security" policy denying access to all unexpected input packets.

3.1. Rules for Balanced Security Policy

These are an example set of generic rules to be applied. Each would normally be configurable, either by the user directly or on behalf of the user by a subscription service. This document does not address the statefulness of the filtering rules as its main objective is to present an approach where some protocols (identified by layer-4 ports) are assumed weak or malevolent and therefore are blocked while all other protocols are assumed benevolent and are permitted.

If we name all nodes on the residential side of the CPE as ’inside’ and all nodes on the Internet as ’outside’, and any packet sent from outside to inside as being ’inbound’ and ’outbound’ in the other direction, then the behavior of the CPE is described by a small set or rules:

1. Rule RejectBogon: apply ingress filtering in both directions per [RFC3704] and [RFC2827] for example with unicast reverse path forwarding (uRPF) checks (anti-spoofing) for all inbound and outbound traffic (implicitly blocking link-local and ULA in the same shot), as described in Section 2.1 Basic Sanitation and Section 3.1 Stateless Filters of [RFC6092];

2. Rule AllowManagement: if the CPE is managed by the SP, then allow the management protocols (SSH, SNMP, syslog, TR-069, IPfix, ...) from/to the SP Network Operation Center;

3. Rule ProtectWeakServices: drop all inbound and outbound packets whose layer-4 destination is part of a limited set (see Section 3.2), the intent is to protect against the most common unauthorized access and avoid propagation of worms; an advanced residential user should be able to modify this pre-defined list;

4. Rule Openess: allow all unsolicited inbound packets with rate limiting the initial packet of a new connection (such as TCP SYN, SCTP INIT or DCCP-request, not applicable to UDP) to provide very basic protection against SYN port and address scanning attacks. All transport protocols and all non-deprecated extension headers are accepted. This is a the major deviation from REC-11, REC-17 and REC-33 of [RFC6092].

5. All requirements of [RFC6092] except REC-11, REC-18 and REC-33 must be supported.

3.2. Rules Example for Layer-4 Protection: Swisscom Implementation

As of 2013, Swisscom has implemented the rule ProtectWeakService as described below. This is meant as an example and must not be followed blindly: each implementer has specific needs and requirements. Furthermore, the example below will not be updated as time passes, whereas threats will evolve.

Transport	Port	Description
tcp	22	Secure Shell (SSH)
tcp	23	Telnet
tcp	80	HTTP
tcp	3389	Microsoft Remote Desktop Protocol
tcp	5900	VNC remote desktop protocol

Table 1: Drop Inbound

Transport	Port	Description
tcp-udp	88	Kerberos
tcp	111	SUN Remote Procedure Call
tcp	135	MS Remote Procedure Call
tcp	139	NetBIOS Session Service
tcp	445	Microsoft SMB Domain Server
tcp	513	Remote Login
tcp	514	Remote Shell
tcp	548	Apple Filing Protocol over TCP
tcp	631	Internet Printing Protocol
udp	1900	Simple Service Discovery Protocol
tcp	2869	Simple Service Discovery Protocol
udp	3702	Web Services Dynamic Discovery
udp	5353	Multicast DNS
udp	5355	Link-Lcl Mcast Name Resolution

```
+-----------+------+------------------------------------+
```

Table 2: Drop Inbound and Outbound

Choosing services to protect is not an easy task, and as of 2013 there is no public service proposing a list of ports to use in such a policy. The Swisscom approach was to think in terms of services, by defining a list of services that are LAN-Only (ex: Multicast DNS) whose communication is denied by the policy both inbound and outbound, and a list of services that are known to be weak or vulnerable like management protocols that could be activated unbeknownst to the user.

The process used to set-up and later update the filters is out of scope of this document. The update of the specific rules could be done together with a firmware upgrade or by a policy update (for example using Broadband Forum TR-069).

Among other sources, [DSHIELD] was used by Swisscom to set-up their filters. Another source of information could be the appendix A of [TR124]. The L4-filter as described does not block GRE tunnels ([RFC2473]) so this is a deviation from [RFC6092].

Note: the authors believe that with a dozen of rules only, a naive and unaware residential subscriber would be reasonably protected. Of course, technically-aware susbcribers should be able to open other applications (identified by their layer-4 ports or IP protocol numbers) through their CPE using some kind of user interface or even to select a completely different security policy such as the open or ’closed’ policies defined by [RFC6092]. This is the case in the Swisscom deployment.

It is worth mentioning that PCP ([RFC6887]), UPnP ([IGD]) and similar protocols can also be used to dynamically override the default rules.

4. IANA Considerations

There are no extra IANA consideration for this document.

5. Security Considerations

The security policy protects from the following type of attacks:

- Unauthorized access because vulnerable ports are blocked

Depending on the extensivity of the filters, certain vulnerabilities could be protected or not. It does not preclude the need for end-devices to have proper host-protection as most of those devices (smartphones, laptops, etc.) would anyway be exposed to completely unfiltered internet at some point of time. The policy addresses the major concerns related to the loss of stateful filtering imposed by IPV4 NAPT when enabling public globally reachable IPv6 in the home.

To the authors’ knowledge, there has not been any incident related to this deployment in Swisscom network, and no customer complaints have been registered.

This set of rules cannot help with the following attacks:

- Flooding of the CPE access link;
- Malware which is fetched by inside hosts on a hostile web site (which is in 2013 the majority of infection sources).

6. Acknowledgements

The authors would like to thank several people who initiated the discussion on the ipv6-ops@lists.cluenet.de mailing list and others who provided us valuable feedback and comments, notably: Tore Anderson, Rajiv Asati, Fred Baker, Lorenzo Colitti, Paul Hoffman, Merike Kaeo, Simon Leinen, Eduard Metz, Martin Millnert, Benedikt Stockebrand. Thanks as well to the following SP that discussed with the authors about this technique: Altibox, Swisscom and Telenor.

7. Informative References

[DSHIELD] DShield, "Port report: DShield", <https://secure.dshield.org/portreport.html?sort=records>.

[IGD] UPnP Forum, "WANIPConnection:2 Service", December 20110, <http://upnp.org/specs/gw/UPnP-gw-WANIPConnection-v2-Service.pdf>.

[RFC2473] Conta, A. and S. Deering, "Generic Packet Tunneling in IPv6 Specification", RFC 2473, December 1998.

[RFC2827] Ferguson, P. and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", BCP 38, RFC 2827, May 2000.

[RFC3704] Baker, F. and P. Savola, "Ingress Filtering for Multihomed Networks", BCP 84, RFC 3704, March 2004.

[RFC6092] Woodyatt, J., "Recommended Simple Security Capabilities in Customer Premises Equipment (CPE) for Providing Residential IPv6 Internet Service", RFC 6092, January 2011.

[RFC6583] Gashinsky, I., Jaeggli, J., and W. Kumari, "Operational Neighbor Discovery Problems", RFC 6583, March 2012.

[RFC6887] Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", RFC 6887, April 2013.

[TR124] Broadband Forum, "Functional Requirements for Broadband Residential Gateway Devices", December 2006, <http://www.broadband-forum.org/technical/download/TR-124.pdf>.

Authors' Addresses

Martin Gysi
Swisscom
Binzring 17
Zuerich 8045
Switzerland

Phone: +41 58 223 57 24
Email: Martin.Gysi@swisscom.com

Guillaume Leclanche
Viagenie
246 Aberdeen
Quebec, QC G1R 2E1
Canada

Phone: +1 418 656 9254
Email: guillaume.leclanche@viagenie.ca

Eric Vyncke (editor)
Cisco Systems
De Kleetlaan 6a
Diegem 1831
Belgium

Phone: +32 2 778 4677
Email: evyncke@cisco.com

Ragnar Anfinsen
Altibox
Breiflaatveien 18
Stavanger 4069
Norway

Phone: +47 93488235
Email: Ragnar.Anfinsen@altibox.no

v6ops
Internet-Draft
Intended status: Informational
Expires: August 7, 2014

D. Lopez
Telefonica I+D
Z. Chen
China Telecom
T. Tsou
Huawei Technologies (USA)
C. Zhou
Huawei Technologies
A. Servin
LACNIC

February 3, 2014

IPv6 Operational Guidelines for Datacenters
draft-ietf-v6ops-dc-ipv6-01

Abstract

This document is intended to provide operational guidelines for datacenter operators planning to deploy IPv6 in their infrastructures. It aims to offer a reference framework for evaluating different products and architectures, and therefore it is also addressed to manufacturers and solution providers, so they can use it to gauge their solutions. We believe this will translate in a smoother and faster IPv6 transition for datacenters of these infrastuctures.

The document focuses on the DC infrastructure itself, its operation, and the aspects related to DC interconnection through IPv6. It does not consider the particular mechanisms for making Internet services provided by applications hosted in the DC available through IPv6 beyond the specific aspects related to how their deployment on the Data Center (DC) infrastructure.

Apart from facilitating the transition to IPv6, the mechanisms outlined here are intended to make this transition as transparent as possible (if not completely transparent) to applications and services running on the DC infrastructure, as well as to take advantage of IPv6 features to simplify DC operations, internally and across the Internet.

Status of this Memo

This Internet-Draft will expire on August 7, 2014.

Copyright Notice

Table of Contents

1. Introduction

The need for considering the aspects related to IPv4-to-IPv6 transition for all devices and services connected to the Internet has been widely mentioned elsewhere, and it is not our intention to make an additional call on it. Just let us note that many of those services are already or will soon be located in Data Centers (DC), what makes considering the issues associated to DC infrastructure transition a key aspect both for these infrastructures themselves, and for providing a simpler and clear path to service transition.

All issues discussed here are related to DC infrastructure transition, and are intended to be orthogonal to whatever particular mechanisms for making the services hosted in the DC available through IPv6 beyond the specific aspects related to their deployment on the infrastructure. General mechanisms related to service transition have been discussed in depth elsewhere (see, for example [RFC6883] and [I-D.ietf-v6ops-enterprise-incremental-ipv6]) and are considered to be independent to the goal of this discussion. The applicability of these general mechanisms for service transition will, in many cases, depend on the supporting DC's infrastructure characteristics. However, this document intends to keep both problems (service vs. infrastructure transition) as different issues.

Furthermore, the combination of the regularity and controlled management in a DC interconnection fabric with IPv6 universal end-to-end addressing should translate in simpler and faster VM migrations, either intra- or inter-DC, and even inter-provider.

2. Architecture and Transition Stages

This document presents a transition framework structured along transition stages and operational guidance associated with the degree of penetration of IPv6 into the DC communication fabric. It is worth noting we are using these stages as a classification mechanism, and they have not to be associated with any a succession of steps from a v4-only infrastructure to full-fledged v6, but to provide a framework that operators, users, and even manufacturers could use to assess their plans and products.

There is no (explicit or implicit) requirement on starting at the stage describe in first place, nor to follow them in successive order. According to their needs and the available solutions, DC operators can choose to start or remain at a certain stage, and freely move from one to another as they see fit, without contravening this document. In this respect, the classification intends to support the planning in aspects such as the adaptation of the different transition stages to the evolution of traffic patterns, or risk assessment in what relates to deploying new components and incorporating change control, integration and testing in highly-complex multi-vendor infrastructures.

Three main transition stages can be considered when analyzing IPv6 deployment in the DC infrastructure, all compatible with the availability of services running in the DC through IPv6:

- Experimental. The DC keeps a native IPv4 infrastructure, with gateway routers (or even application gateways when services require so) performing the adaptation to requests arriving from the IPv6 Internet.
- Dual stack. Native IPv6 and IPv4 are present in the infrastructure, up to whatever the layer in the interconnection scheme where L3 is applied to packet forwarding.
- IPv6-Only. The DC has a fully pervasive IPv6 infrastructure, including full IPv6 hypervisors, which perform the appropriate tunneling or NAT if required by internal applications running IPv4.

2.1. General Architecture

The diagram in Figure 1 depicts a generalized interconnection schema in a DC.

```
             |               |
       +-----+-----+   +-----+-----+
       |  Gateway  |   |  Gateway  |          Internet / Remote Access
       +-----+-----+   +-----+-----+                  Modules
             |               |
             +---+-----------+
                 |     |
         +---+---+     +---+---+
         | Core0 |     | CoreN |              Core
         +---+---+     +---+---+
               /  \    /      /
              /    \-----\   /
             /   /---/    \ /
           +--------+       +--------+
         +/-------+ |     +/-------+ |
         | Aggr01 | +-----| AggrN1 | +        Aggregation
         +---+---+/       +--------+/
          /     \         /      \
         /       \       /        \
   +-----+    +-----+   +-----+    +-----+
   | T11 |... | T1x |   | T21 |... | T2y |    Access
   +-----+    +-----+   +-----+    +-----+
   | HyV |    | HyV |   | HyV |    | HyV |    Physical Servers
   +:::::+    +:::::+   +:::::+    +:::::+
   | VMs |    | VMs |   | VMs |    | VMs |    Virtual Machines
   +-----+    +-----+   +-----+    +-----+
   . . . .    . . . .   . . . .    . . . .
   +-----+    +-----+   +-----+    +-----+
   | HyV |    | HyV |   | HyV |    | HyV |
   +:::::+    +:::::+   +:::::+    +:::::+
   | VMs |    | VMs |   | VMs |    | VMs |
   +-----+    +-----+   +-----+    +-----+
```

Figure 1: DC Interconnnection Schema

- Hypervisors provide connection services (among others) to virtual machines running on physical servers.

- Access elements provide connectivity directly to/from physical servers. The access elements are typically placed either top-of-rack (ToR) or end-of-row(EoR).

- Aggregation elements group several (many) physical racks to achieve local integration and provide as much structure as possible to data paths.

- Core elements connect all aggregation elements acting as the DC backbone.

- One or several gateways connecting the DC to the Internet, Branch Offices, Partners, Third-Parties, and/or other DCs. The interconnectivity to other DC may be in the form of VPNs, WAN links, metro links or any other form of interconnection.

In many actual deployments, depending on DC size and design decisions, some of these elements may be combined (core and gateways are provider by the same routers, or hypervisors act as access elements) or virtualized to some extent, but this layered schema is the one that best accommodates the different options to use L2 or L3 at any of the different DC interconnection layers, and will help us in the discussion along the document.

2.2. Experimental Stage. Native IPv4 Infrastructure

This transition stage corresponds to the first step that many datacenters may take (or have taken) in order to make their external services initially accessible from the IPv6 Internet and/or to evaluate the possibilities around it, and corresponds to IPv6 traffic patterns totally originated out of the DC or their tenants, being a small percentage of the total external requests. At this stage, DC network scheme and addressing do not require any important change, if any.

It is important to remark that in no case this can be considered a permanent stage in the transition, or even a long-term solution for incorporating IPv6 into the DC infrastructure. This stage is only recommended for experimentation or early evaluation purposes.

The translation of IPv6 requests into the internal infrastructure addressing format occurs at the outmost level of the DC Internet connection. This can be typically achieved at the DC gateway routers, that support the appropriate address translation mechanisms for those services required to be accessed through native IPv6 requests. The policies for applying adaptation can range from performing it only to a limited set of specified services to providing a general translation service for all public services. More granular mechanisms, based on address ranges or more sophisticated dynamic policies are also possible, as they are applied by a limited set of control elements. These provide an additional level of control to the usage of IPv6 routable addresses in the DC environment, which can be especially significant in the experimentation or early deployment phases this stage is applicable to.

Even at this stage, some implicit advantages of IPv6 application come into play, even if they can only be applied at the ingress elements:

- Flow labels can be applied to enhance load distribution, as described in [RFC7098]. If the incoming IPv6 requests are adequately labeled the gateway systems can use the flow labels as a hint for applying load-balancing mechanisms when translating the requests towards the IPv4 internal network.

- During VM migration (intra- or even inter-DC), Mobile IPv6 mechanisms can be applied to keep service availability during the transient state.

2.2.1. Off-shore v6 Access

This model is also suitable to be applied in an "off-shore" mode by the service provider connecting the DC infrastructure to the Internet, as described in [I-D.sunq-v6ops-contents-transition].

When this off-shore mode is applied, the original source address will be hidden to the DC infrastructure, and therefore identification techniques based on it, such as geolocation or reputation evaluation, will be hampered. Unless there is a specific trust link between the DC operator and the ISP, and the DC operator is able to access equivalent identification interfaces provided by the ISP as an additional service, the off-shore experimental stage cannot be considered applicable when source address identification is required.

2.3. Dual Stack Stage. Internal Adaptation

This stage requires dual-stack elements in some internal parts of the DC infrastructure. This brings some degree of partition in the infrastructure, either in a horizontal (when data paths or management interfaces are migrated or left in IPv4 while the rest migrate) or a vertical (per tenant or service group), or even both.

Although it may seem an artificial case, situations requiring this stage can arise from different requirements from the user base, or the need for technology changes at different points of the infrastructure, or even the goal of having the possibility of experimenting new solutions in a controlled real-operations environment, at the price of the additional complexity of dealing with a double protocol stack, as noted in [RFC6883] and elsewhere.

This transition stage can accommodate different traffic patterns, both internal and external, though it better fits to scenarios of a clear differentiation of different types of traffic (external vs. internal, data vs management...), and/or a more or less even distribution of external requests. A common scenario would include native dual stack servers for certain services combined with single stack ones for others (web server in dual stack and database servers only supporting v4, for example).

At this stage, the advantages outlined above on load balancing based on flow labels and Mobile IP mechanisms are applicable to any L3-based mechanism (intra- as well as inter-DC). They will translate into enhanced VM mobility, more effective load balancing, and higher service availability. Furthermore, the simpler integration provided by IPv6 to and from the L2 flat space to the structured L3 one can be applied to achieve simpler deployments, as well as alleviating encapsulation and fragmentation issues when traversing between L2 and L3 spaces. With an appropriate prefix management, automatic address assignment, discovery, and renumbering can be applied not only to public service interfaces, but most notably to data and management paths. Other potential advantages include the application of multicast scopes to limit broadcast floods, and the usage of specific security headers to enhance tenant differentiation.

In general, all these advantages are especially significative to overlay techniques applied to support multi-tenancy and inter-DC operation.

On the other hand, this stage requires a much more careful planning of addressing (please refer to ([RFC5375]) schemas and access control, according to security levels. While the experimental stage implies relatively few global routable addresses, this one brings the advantages and risks of using different kinds of addresses at each point of the IPv6-aware infrastructure.

2.3.1. Dual-stack at the Aggregation Layer

```
    +---------------------+
    | Internet / External |
    +---------+-----------+
              |
        +-----+----+
        |  Gateway |
        +-----+----+
              .
              .           Core Level
              .
           +--+--+
           | FW  |
           +--+--+
              |           Aggregation Level
           +--+--+
           | MB  |
           +--+--+
           _ / \_
          /       \
    +--+--+     +--+--+
    | Web | ... | Web |
    +--+--+     +--+--+
       | \ __ _ _/ |
       | /       \ |
    +--+--+     +--+--+
    |Cache|     | DB  |
    +-----+     +-----+
```

Figure 2: Data Center Application Scheme

An initial approach corresponding to this transition stage relies on taking advantage of specific elements at the aggregation layer described in Figure 1, and make them able to provide dual-stack gatewaying to the IPv4-based servers and data infrastructure.

Typically, firewalls (FW) are deployed as the security edge of the whole service domain and provides safe access control of this service domain from other function domains. In addition, some application optimization based on devices and security devices (generally known as middleboxes, e.g. Load Balancers, SSL VPN, IPS and etc.) may be deployed in the aggregation level to alleviate the burden of the server and to guarantee deep security, as shown in Figure 2. The choice of a particular kind of middlebox for this dual-stack approach shall be based on the nature of the services and the deployment of the middleboxes in the DC infrastructure.

The middlebox could be upgraded to support the data transmission. There may be two ways to achieve this at the edge of the DC: Encapsulation and NAT. In the encapsulation case, the middlebox function carries the IPv6 traffic over IPv4 using an encapsulation (IPv6-in-IPv4). In the NAT case, there are already some technologies to solve this problem. For example, DNS and NAT devices could be concatenated for IPv4/IPv6 translation if IPv6 host needs to visit IPv4 servers. However, this may require the concatenation of multiple network devices, which means the NAT tables needs to be synchronized at different devices. As described below, a simplified IPv4/IPv6 translation model can be applied, which could be implemented in the device. The mapping information of IPv4 and IPv6 will be generated automatically based on the information of the middlebox. The host IP address will be translated without port translation.

```
                        +----------+------------------------------+
                        |Dual Stack| IPv4-only       +----------+ |
                        |          |            +----|Web Server| |
                        |   +------|------+    /     +----------+ |
+--------+  +-------+   |   |      |      |   /                   |
|Internet|--|Gateway|---|---+Load-Balancer+-- \                   |
|        |  |       |   |   |      |      |    \     +----------+ |
+--------+  +-------+   |   +------|------+     +----|Web Server| |
                        |          |                 +----------+ |
                        +----------+------------------------------+
```

Figure 3: Dual Stack middlebox (Load-Balancer) mechanism

As shown in Figure 3,the middlebox (a load-balancer, LB, in this case) can be considered divided into two parts: The dual-stack part facing the external border, and the IPv4-only part which contains the traditional LB functions. The IPv4 DC is allocated an IPv6 prefix which is for the VSIPv6 (Virtual Service IPv6 Address). We suggest that the IPv6 prefix is not the well-known prefix in order to avoid the IPv4 routings of the services in different DCs spread to the IPv6 network. The VSIPv4 (Virtual Service IPv4 Address) is embedded in VSIPv6 using the allocated IPv6 prefix. In this way, the LB has the stateless IP address mapping between VSIPv6 and VSIPv4, and synchronization is not required between LB and DNS64 server.

The dual-stack part of the LB has a private IPv4 address pool. When IPv6 packets arrive, the dual-stack part does the one-on-one SIP (source IP address) mapping (as defined in [I-D.sunq-v6ops-contents-transition]) between IPv4 private address and IPv6 SIP. Because there will be too many UDP/TCP sessions between the DC and Internet, the IP addresses binding tables between

IPv6 and IPv4 are not session-based, but SIP-based. Thus, the dual-stack part of LB builds IP binding stateful tables for the host IPv6 address and private IPv4 address of the pool. When the following IPv6 packets of the host come from Internet to the LB, the dual stack part does the IP address translation for the packets. Thus, the IPv6 packets were translated to IPv4 packets and sent to the IPv4 only part of the LB.

2.3.2. Dual-stack Extended OS/Hypervisor

Another option for deploying a infrastructure at the dual-stack stage would bring dual-stack much closer to the application servers, by requiring hypervisors, VMs and applications in the v6-capable zone of the DC to be able to operate in dual stack. This way, incoming connections would be dealt in a seamless manner, while for outgoing ones an OS-specific replacement for system calls like gethostbyname() and getaddrinfo() would accept a character string (an IPv4 literal, an IPv6 literal, or a domain name) and would return a connected socket or an error message, having executed a happy eyeballs algorithm ([RFC6555]).

If these hypothetical system call replacements were smart enough, they would allow the transparent interoperation of DCs with different levels of v6 penetration, either horizontal (internal data paths are not migrated, for example) or vertical (per tenant or service group). This approach requires, on the other hand, all the involved DC infrastructure to become dual-stack, as well as some degree of explicit application adaptation.

2.4. IPv6-Only Stage. Pervasive IPv6 Infrastructure

We can consider a DC infrastructure at the final stage when all network layer elements, including hypervisors, are IPv6-aware and apply it by default. Conversely with the experimental stage, access from the IPv4 Internet is achieved, when required, by protocol translation performed at the edge infrastructure elements, or even supplied by the service provider as an additional network service.

There are different drivers that could motivate DC managers to transition to this stage. In principle the scarcity of IPv4 addresses may require to reclaim IPv4 resources from portions of the network infrastructure which no longer need them. Furthermore, the unavailability of IPv4 address would make dual-stack environments not possible anymore and careful assessments will be perfumed to asses where to use the remaining IPv4 resources.

Another important motivation to move DC operations from dual-stack to IPv6-only is to save costs and operation activities that managing a single-stack network could bring in comparison with managing two stacks. Today, besides of learning to manage two different stacks, network and system administrators require to duplicate other tasks such as IP address management, firewalls configuration, system security hardening and monitoring among others. These activities are not just costly for the DC management, they may also may lead to configuration errors and security holes. In particular, a few activities have special impact on costs for dual-stacked infrastructures:

- Development. When a new device or app version is released, it must be tested three times: IPv4, dual-stack, and IPv6-only. Though this does not imply a triple the effort once the development environment is set up, a general estimate is that it implies a 10% additional cost.

- Test. Everything QA procedure must be performed at least twice and in many cases three times, with an estimate 10% incremental effort.

- Operation and troubleshooting. While for L1/L2 problems we would be talking of 1% incremental effort (in a few words, once ping6 works, checking ping is very little effort), for L3 problems a rough estimate would an increment of 5%.

- Application development. Many applications would require to keep two branches, with a 10-30% additional cost. The estimate here implies a higher range, as applications cover a wide variety of cases.

- Addition on new L3 devices, that should handle IPv4 and IPv6 flows, and provide higher performance to deal with both at the same time. It comes with a cost increment of 5-10%.

- Network management. The incremental costs of managing two L3 network plane would come at around a 10% incremental cost.

In summary, a full dual-stack datacenter would come at an additional 5-10% operating cost than a single-stack one.

This stage can be also of interest for new deployments willing to apply a fresh start aligned with future IPv6 widespread usage, when a relevant amount of requests are expected to be using IPv6, or to take advantage of any of the potential benefits that an IPv6 support infrastructure can provide. Other, and probably more compelling in many cases, drivers for this stage may be either a lack of enough IPv4 resources (whether private or globally unique) or a need to reclaim IPv4 resources from portions of the network which no longer need them. In these circumstances, a careful evaluation of what still needs to speak IPv4 and what does not will need to happen to ensure judicious use of the remaining IPv4 resources.

The potential advantages mentioned for the previous stages (load distribution based on flow labels, mobility mechanisms for transient states in VM or data migration, controlled multicast, and better mapping of L2 flat space on L3 constructs) can be applied at any layer, even especially tailored for individual services. Obviously, the need for a careful planning of address space is even stronger here, though the centralized protocol translation services should reduce the risk of translation errors causing disruptions or security breaches.

[V6DCS] proposes an approach to a next generation DC deployment, already demonstrated in practice, and claims the advantages of materializing the stage from the beginning, providing some rationale for it based on simplifying the transition process. It relies on stateless NAT64 ([RFC6052], [RFC6145]) to enable access from the IPv4 Internet.

2.4.1. Overlay and Chaining Support

A DC infrastructure in this final stage is in the position of providing a much better support to requirements that have been recently formulated, mostly in the scope of other recently created IETF working groups.

In particular, support for highly scalable VPN and multi-tenancy according to the key requirements defined in [I-D.ietf-nvo3-overlay-problem-statement]:

- Traffic isolation, so that a tenant’s traffic is not visible to any other tenant.

- Address independence, so that one tenant’s addressing scheme does not collide with other tenant’s addressing schemes or with addresses used within the data center itself.

- Support the placement and migration of VMs anywhere within the data center, without being limited by DC network constraints such as the IP subnet boundaries of the underlying DC network.

With a pervasive IPv6 infrastructure, these goals can be achieved by means of native addressing and direct interaction of the applications with the network infrastructure of the datacenter, and across multiple datacenters connected via WAN links. Virtual networks can be constructed by a natural consequence of addressing rules, traffic isolation guaranteed by routing mechanisms, and migration directly supported by signaling protocols.

On the other hand, service chaining is consolidating as a technique for dynamically structuring network services, adapting them to user requirements, provider policies, and network state. In this model, service functions, whether physical or virtualized, are not required to reside on the direct data path and traffic is instead steered through required service functions, wherever they are deployed [I-D.ietf-sfc-problem-statement].

Service function chaining requires packets in a given flow intended to follow a particular path to be tagged by a classifier, so intermediate service nodes in the path can route them accordingly. The usage of flow labels can greatly simplify this classification and allow a much simpler deployment of service function chains. Furthermore, it offers much richer possibilities for network architects building chains and paths inside them as well as to application developers willing to get advantage of service chaining, since it provides the possibility of providing rich metadata for any given flow, in a generalization of the use cases described in [RFC6294] and [RFC7098].

2.5. Other Operational Considerations

In this section we review some operation considerations related addressing and management issues in V6 DC infrastructure.

2.5.1. Addressing

There are different considerations related on IPv6 addressing topics in DC. Many of these considerations are already documented in a variety of IETF documents and in general the recommendations and best practices mentioned on them apply in IPv6 DC environments. However we would like to point out some topics that we consider important to mention.

The first question that DC managers often have is the type of IPv6 address to use; that is Provider Aggregated (PA), Provider Independent (PI) or Unique Local IPv6 Addresses (ULAs) [RFC4193] Related to the use of PA vs. PI, we concur with [RFC6883] and [I-D.ietf-v6ops-enterprise-incremental-ipv6] that PI provides independence from the ISP and decreases renumbering issues, it may bring up other considerations as a fee for the allocation, a request process and allocation maintenance to the Regional Internet Registry, etc. In this respect, there is not a specific recommendation to use either PI vs. PA as it would depend also on business and management factors rather than pure technical.

ULAs should be used only in DC infrastructure that does not require access to the public Internet; such devices may be databases servers, application-servers, and management interfaces of webservers and network devices among others. This practice may decrease the renumbering issues when PA addressing is used, as only public faced devices would require an address change. Also we would like to know that although ULAs may provide some security the main motivation for it used should be address management.

Another topic to discuss is the length of prefixes within the DC. In general we recommend the use of subnets of 64 bits for each VLAN or network segment used in the DC. Although subnet with prefixes longer than 64 bits may work, it is necessary that the reader understands that this may break stateless autoconfiguration and at least manual configuration must be employed. For details please read [RFC5375].

Address plans should follow the principles of being hierarchical and able to aggregate address space. We recommend at least to have a /48 for each data-center. If the DC provides services that require subassigment of address space we do not offer a single recommendation (i.e. request a /40 prefix from an RIR or ISP and assign /48 prefixes to customers), as this may depend on other no technical factors. Instead we refer the reader to [RFC6177].

For point-to-point links please refer to the recommendations in [RFC6164].

2.5.2. Management Systems and Applications

Data-centers may use Internet Protocol address management (IPAM) software, provisioning systems and other variety of software to document and operate. It is important that these systems are prepared and possibly modified to support IPv6 in their data models. In general, if IPv6 support for these applications has not been previously done, changes may take sometime as they may be not just adding more space in input fields but also modifying data models and data migration.

2.5.3. Monitoring and Logging

Monitoring and logging are critical operations in any network environment and they should be carried at the same level for IPv6 and IPv4. Monitoring and management operations in V6 DC are by no means different than any other IPv6 networks environments. It is important to consider that the collection of information from network devices is orthogonal to the information collected. For example it is possible to collect data from IPv6 MIBs using IPv4 transport. Similarly it is possible to collect IPv6 data generated by Netflow9/

IPFIX agents in IPv4 transport. In this way the important issue to address is that agents (i.e. network devices) are able to collect data specific to IPv6.

And as final note on monitoring, although IPv6 MIBs are supported by SNMP versions 1 and 2, we recommend to use SNMP version 3 instead.

2.5.4. Costs

It is very possible that moving from a single stack data-center infrastructure to any of the IPv6 stages described in this document may incur in capital expenditures. This may include but it is not confined to routers, load-balancers, firewalls and software upgrades among others. However the cost that most concern us is operational. Moving the DC infrastructure operations from a single-stack to a dual-stack may infer in a variety of extra costs such as application development and testing, operational troubleshooting and service deployment. At the same time, this extra cost may be seeing as saving when moving from a dual-stack DC to an IPv6-Only DC.

Depending of the complexity of the DC network, provisioning and other factors we estimate that the extra costs (and later savings) may be around between 15 to 20%.

2.6. Security Considerations

A thorough collection of operational security aspects for IPv6 network is made in [I-D.ietf-opsec-v6]. Most of them, with the probable exception of those specific to residential users, are applicable in the environment we consider in this document.

2.6.1. Neighbor Discovery Protocol attacks

The first important issue that V6 DC manager should be aware is the attacks against Neighbor Discovery Protocol [RFC6583]. This attack is similar to ARP attacks [RFC4732] in IPv4 but exacerbated by the fact that the common size of an IPv6 subnet is /64. In principle an attacker would be able to fill the Neighbor Cache of the local router and starve its memory and processing resources by sending multiple ND packets requesting information of non-existing hosts. The result would be the inability of the router to respond to ND requests, to update its Neighbor Cache and even to forward packets. The attack does need to be launched with malicious purposes; it could be just the result of bad stack implementation behavior.

R[RFC6583] mentions some options to mitigate the effects of the attacks against NDP. For example filtering unused space, minimizing subnet size when possible, tuning rate limits in the NDP queue and to rely in router vendor implementations to better handle resources and to prioritize NDP requests.

2.6.2. Addressing

Other important security considerations in V6 DC are related to addressing. Because of the large address space is commonly thought that IPv6 is not vulnerable to reconnaissance techniques such as scanning. Although that may be true to force brute attacks, [I-D.ietf-opsec-ipv6-host-scanning] shows some techniques that may be employed to speed up and improve results in order to discover IPv6 address in a subnet. The use of virtual machines and SLACC aggravate this problem due the fact that they tent to use automatically-generated MAC address well known patterns.

To mitigate address-scanning attacks it is recommended to avoid using SLAAC and if used stable privacy-enhanced addresses [I-D.ietf-6man-stable-privacy-addresses] should be the method of address generation. Also, for manually assigned addresses try to avoid IID low-byte address (i.e. from 0 to 256), IPv4-based addresses and wordy addresses especially for infrastructure without a fully qualified domain name.

In spite of the use of manually assigned addresses is the preferred method for V6 DC, SLACC and DHCPv6 may be also used for some special reasons. However we recommend paying special attention to RA [RFC6104] and DHCP [I-D.ietf-opsec-dhcpv6-shield] hijack attacks. In these kinds of attacks the attacker deploys rogue routers sending RA messages or rogue DHCP servers to inject bogus information and possibly to perform a man in the middle attack. In order to mitigate this problem it is necessary to apply some techniques in access switches such as RA-Guard [RFC6105] at least.

Another topic that we would like to mention related to addressing is the use of ULAs. As we previously mentioned, although ULAs may be used to hide host from the outside world we do not recommend to rely on them as a security tool but better as a tool to make renumbering easier.

2.6.3. Edge filtering

In order to avoid being used as a source of amplification attacks is it important to follow the rules of BCP38 on ingress filtering. At the same time it is important to filter-in on the network border all the unicast traffic and routing announcement that should not be routed in the Internet, commonly known as "bogus prefixes".

2.6.4. Final Security Remarks

Finally, let us just emphasize the need for careful configuration of access control rules at the translation points. This latter one is specially sensitive in infrastructures at the dual-stack stage, as the translation points are potentially distributed, and when protocol translation is offered as an external service, since there can be operational mismatches.

2.7. IANA Considerations

None.

2.8. Acknowledgements

We would like to thank Tore Anderson, Wes George, Ray Hunter, Joel Jaeggli, Fred Baker, Lorenzo Colitti, Dan York, Carlos Martinez, Lee Howard, Alejandro Acosta, Alexis Munoz, Nicolas Fiumarelli, Santiago Aggio and Hans Velez for their questions, suggestions, reviews and comments.

3. Informative References

[I-D.ietf-6man-stable-privacy-addresses]
Gont, F., "A Method for Generating Semantically Opaque Interface Identifiers with IPv6 Stateless Address Autoconfiguration (SLAAC)", draft-ietf-6man-stable-privacy-addresses-17 (work in progress), January 2014.

[I-D.ietf-nvo3-overlay-problem-statement]
Narten, T., Gray, E., Black, D., Fang, L., Kreeger, L., and M. Napierala, "Problem Statement: Overlays for Network Virtualization", draft-ietf-nvo3-overlay-problem-statement-04 (work in progress), July 2013.

[I-D.ietf-opsec-dhcpv6-shield]
Gont, F., Will, W., and G. Velde, "DHCPv6-Shield: Protecting Against Rogue DHCPv6 Servers", draft-ietf-opsec-dhcpv6-shield-02 (work in progress), February 2014.

[I-D.ietf-opsec-ipv6-host-scanning]
Gont, F. and T. Chown, "Network Reconnaissance in IPv6 Networks", draft-ietf-opsec-ipv6-host-scanning-03 (work in progress), January 2014.

[I-D.ietf-opsec-v6]
Chittimaneni, K., Kaeo, M., and E. Vyncke, "Operational Security Considerations for IPv6 Networks", draft-ietf-opsec-v6-04 (work in progress), October 2013.

[I-D.ietf-sfc-problem-statement]
Quinn, P. and T. Nadeau, "Service Function Chaining Problem Statement", draft-ietf-sfc-problem-statement-00 (work in progress), January 2014.

[I-D.ietf-v6ops-enterprise-incremental-ipv6]
Chittimaneni, K., Chown, T., Howard, L., Kuarsingh, V., Pouffary, Y., and E. Vyncke, "Enterprise IPv6 Deployment Guidelines", draft-ietf-v6ops-enterprise-incremental-ipv6-05 (work in progress), January 2014.

[I-D.sunq-v6ops-contents-transition]
Sun, Q., Liu, D., Zhao, Q., Liu, Q., Xie, C., Li, X., and J. Qin, "Rapid Transition of IPv4 contents to be IPv6-accessible", draft-sunq-v6ops-contents-transition-03 (work in progress), March 2012.

[RFC4193] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast Addresses", RFC 4193, October 2005.

[RFC4732] Handley, M., Rescorla, E., and IAB, "Internet Denial-of-Service Considerations", RFC 4732, December 2006.

[RFC5375] Van de Velde, G., Popoviciu, C., Chown, T., Bonness, O., and C. Hahn, "IPv6 Unicast Address Assignment Considerations", RFC 5375, December 2008.

[RFC6052] Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", RFC 6052, October 2010.

[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.

[RFC6145] Li, X., Bao, C., and F. Baker, "IP/ICMP Translation Algorithm", RFC 6145, April 2011.

[RFC6164] Kohno, M., Nitzan, B., Bush, R., Matsuzaki, Y., Colitti, L., and T. Narten, "Using 127-Bit IPv6 Prefixes on Inter-Router Links", RFC 6164, April 2011.

[RFC6177] Narten, T., Huston, G., and L. Roberts, "IPv6 Address Assignment to End Sites", BCP 157, RFC 6177, March 2011.

[RFC6294] Hu, Q. and B. Carpenter, "Survey of Proposed Use Cases for the IPv6 Flow Label", RFC 6294, June 2011.

[RFC6555] Wing, D. and A. Yourtchenko, "Happy Eyeballs: Success with Dual-Stack Hosts", RFC 6555, April 2012.

[RFC6583] Gashinsky, I., Jaeggli, J., and W. Kumari, "Operational Neighbor Discovery Problems", RFC 6583, March 2012.

[RFC6883] Carpenter, B. and S. Jiang, "IPv6 Guidance for Internet Content Providers and Application Service Providers", RFC 6883, March 2013.

[RFC7098] Carpenter, B., Jiang, S., and W. Tarreau, "Using the IPv6 Flow Label for Load Balancing in Server Farms", RFC 7098, January 2014.

[V6DCS] "The case for IPv6-only data centres", <https://ripe64.ripe.net/presentations/67-20120417-RIPE64-The_Case_for_IPv6_Only_Data_Centres.pdf>.

Authors’ Addresses

Diego R. Lopez
Telefonica I+D
Don Ramon de la Cruz, 84
Madrid 28006
Spain

Phone: +34 913 129 041
Email: diego@tid.es

Zhonghua Chen
China Telecom
P.R.China

Phone:
Email: 18918588897@189.cn

Tina Tsou
Huawei Technologies (USA)
2330 Central Expressway
Santa Clara, CA 95050
USA

Phone: +1 408 330 4424
Email: Tina.Tsou.Zouting@huawei.com

Cathy Zhou
Huawei Technologies
Bantian, Longgang District
Shenzhen 518129
P.R. China

Phone:
Email: cathy.zhou@huawei.com

Arturo Servin
LACNIC
Rambla Republica de Mexico 6125
Montevideo 11300
Uruguay

Phone: +598 2604 2222
Email: aservin@lacnic.net

V6OPS
Internet-Draft
Intended status: Informational
Expires: February 18, 2017

B. Liu
S. Jiang
Huawei Technologies
X. Gong
W. Wang
BUPT University
E. Rey
ERNW GmbH

August 17, 2016

DHCPv6/SLAAC Interaction Problems on Address and DNS Configuration

draft-ietf-v6ops-dhcpv6-slaac-problem-07

Abstract

The IPv6 Neighbor Discovery (ND) Protocol includes an ICMPv6 Router Advertisement (RA) message. The RA message contains three flags, indicating the availability of address auto-configuration mechanisms and other configuration such as DNS-related configuration. These are the M, O, and A flags, which by definition are advisory, not prescriptive.

This document describes divergent host behaviors observed in popular operating systems. It also discusses operational problems that the divergent behaviors might cause.

Status of This Memo

This Internet-Draft will expire on February 18, 2017.

Copyright Notice

Table of Contents

1. Introduction

IPv6 [RFC2460] hosts could invoke Neighbor Discovery (ND) [RFC4861] to to discover which auto-configuration mechanisms are available to them. There are two auto-configuration mechanisms in IPv6:

- DHCPv6 [RFC3315]

- Stateless Address Autoconfiguration (SLAAC) [RFC4862]

ND specifies an ICMPv6-based [RFC4443] Router Advertisement (RA) message. Routers periodically multicast the RA messages to all on-link nodes. They also unicast RA messages in response to solicitations. The RA message contains (but not limited to):

- an M (Managed) flag, indicating that addresses are available from DHCPv6 or not

- an O (OtherConfig) flag, indicating that other configuration information (e.g., DNS-related information) is available from DHCPv6 or not

- zero or more Prefix Information (PI) Options

    an A (Autonomous) flag is included, indicating that the prefix can be used for SLAAC or not

The M and O flags are advisory, not prescriptive. For example, the M flag indicates that addresses are available from DHCPv6, but It does not indicate that hosts are required to acquire addresses from DHCPv6. Similar statements can be made about the O flag. (A flag is also advisory by definition in standard, but it is quite prescriptive in implementations according to the test results in the appendix.)

Because of the advisory definition of the flags, in some cases different operating systems appear divergent behaviors. This document analyzes possible divergent host behaviors might happen (most of the possible divergent behaviors are already observed in popular operating systems) and the operational problems might caused by divergent behaviors.

2. The M, O and A Flags

This section briefly reviews how the M, O and A flags are defined in ND[RFC4861] and SLAAC[RFC4862].

2.1. Flags Definition

- M (Managed) Flag

  As decribed in [RFC4861], "When set, it indicates that addresses are available via Dynamic Host Configuration Protocol".

- O (Otherconfig) Flag

  "When set, it indicates that other configuration information is available via DHCPv6. Examples of such information are DNS-related information or information on other servers within the network." [RFC4861]

  "If neither M nor O flags are set, this indicates that no information is available via DHCPv6" . [RFC4861]

- A (Autonomous) Flag

  A flag is defined in the PIO, "When set indicates that this prefix can be used for stateless address configuration as specified in [RFC4862].".

2.2. Flags Relationship

Per [RFC4861], "If the M flag is set, the O flag is redundant and can be ignored because DHCPv6 will return all available configuration information.".

There is no explicit description of the relationship between A flag and the M/O flags.

3. Behavior Ambiguity Analysis

The ambiguity of the flags definition means that when interpreting the same messages, different hosts might behave differently. The ambiguity space is analyzed as the following aspects.

1) Dependency between DHCPv6 and RA

   In standards, behavior of DHCPv6 and Neighbor Discovery protocols is specified respectively. But it is not clear that whether there should be any dependency between them. More specifically, it is unclear whether RA (with M=1) is required to trigger DHCPv6; in other words, It is unclear whether hosts should initiate DHCPv6 by themselves if there are no RAs at all.

2) Overlapping configuration between DHCPv6 and RA

When address and DNS configuration are both available from DHCPv6 and RA, it is not clear how to deal with the overlapping information. Should the hosts accept all the information? If the information conflicts, which one should take higher priority?

For DNS configuration, [RFC6106] clearly specifies "In the case where the DNS options of RDNSS and DNSSL can be obtained from multiple sources, such as RA and DHCP, the IPv6 host SHOULD keep some DNS options from all sources" and "the DNS information from DHCP takes precedence over that from RA for DNS queries" (Section 5.3.1 of [RFC6106]). But for address configuration, there’s no such guidance.

3) Interpretation on Flags Transition

- Impact on SLAAC/DHCPv6 on and off

  When flags are in transition, e.g. the host is already SLAAC-configured, then M flag changes from FALSE to TRUE, it is not clear whether the host should start DHCPv6 or not; or vise versa, the host is already configured by both SLAAC and DHCPv6, then M flag change from TRUE to FALSE, it is also not clear whether the host should turn DHCPv6 off or not.

- Impact on address lifetime

  When one address configuration method is off, that is, the A flag or M flag changes from TRUE to FALSE, it is not clear whether one host should immediately release the corresponding address or just retain it until the lifetime expires.

4) Relationship between the Flags

As described above, the relationship between A flag and M/O flags is unspecified.

It could be reasonably deduced that M flag should be independent from A flag. In other words, the M flag only cares DHCPv6 address configuration, while the A flag only cares SLAAC.

But for A flag and O flag, ambiguity could possibly happen. For example, when A is FALSE (when M is also FALSE) and O is TRUE, it is not clear whether the host should initiate a stand-alone stateless DHCPv6 session.

Divergent behaviors on all these aspects have been observed among some popular operating systems as described in Section 4 below.

4. Observed Divergent Host Behaviors

The authors tested several popular operating systems in order to determine what behaviors the M, O and A flag elicit. In some cases, the M, O and A flags elicit divergent behaviors. The table below characterizes those cases. For test details, please refer to Appendix A.

Operation diverges in two ways: one is regarding to address auto-configuration; the other is regarding to DNS configuration.

4.1. Divergent Behavior on Address Auto-Configuration

Divergence 1-1

- Host state: has not acquired any addresses.
- Input: no RA.
- Divergent Behavior
  1) Acquiring addresses from DHCPv6.
  2) No DHCPv6 action.

Divergence 1-2

- Host state: has acquired addresses from DHCPv6 only (M = 1).
- Input: RA with M =0.
- Divergent Behavior
  1) Releasing DHCPv6 addresses immediately.
  2) Releasing DHCPv6 addresses when they expire.

Divergence 1-3

- Host state: has acquired addresses from SLAAC only (A=1).
- Input: RA with M =1.
- Divergent Behavior

1) Acquiring DHCPv6 addresses immediately.

2) Acquiring DHCPv6 addresses only if their SLAAC addresses expire and cannot be refreshed.

4.2. Divergent Behavior on DNS Configuration

Divergence 2-1

- Host state: has not acquired any addresses or information.
- Input: RA with M=0, O=1, no RDNSS; and a DHCPv6 server on the same link providing RDNSS (regardless of address provisioning).
- Divergent Behavior

  1) Acquiring RDNNS from DHCPv6, regardless of the A flag setting.

  2) Acquiring RDNNS from DHCPv6 only if A=1.

Divergence 2-2

(This divergence is only for those operations systems which support[RFC6106].)

- Host state: has not acquired any addresses or information.
- Input: RA with M=0/1, A=1, O=1 and an RDNSS is advertised; and a DHCPv6 server on the same link providing IPv6 addresses and RDNSS.
- Divergent Behavior

  1) Getting RDNSS from both the RAs and the DHCPv6 server, and the RDNSS obtained from the router has a higher priority.

  2) Getting RDNSS from both the RAs and the DHCPv6 server, but the RDNSS obtained from the DHCPv6 server has a higher priority.

  3) Getting RDNSS from the router, and a "domain search list" information only from the DHCPv6 server(no RDNSS).

Divergence 2-3

(This divergence is only for those operations systems which support[RFC6106].)

- Host state: has acquired address and RDNSS from the first router’s RAs (M=0, O=0, PIO with A=1, and RDNSS advertised).

- Input: another router advertising M=1, O=1, no prefix information; and a DHCPv6 server on the same link providing IPv6 addresses and RDNSS.

- Divergent Behavior

  1) Never getting any information (neither IPv6 address nor RDNSS) from the DHCPv6 server.

  2) Getting an IPv6 address and RDNSS from the DHCPv6 server while retaining the address and RDNSS obtained from the RAs of the first router.

     (More details: the RDNSS obtained from the first router has a higher priority; when they receive again RAs from the first router, they lose/forget the information (IPv6 address and RDNSS) obtained from the DHCPv6 server.)

Divergence 2-4

(This divergence is only for those operations systems which support[RFC6106].)

- Host state: has acquired address and RDNSS from the DHCPv6 server indicated by the first router (M=1, O=1, no PIO or RDNSS advertised).

- Input: another router advertising M=0, O=0, PIO with A=1, and RNDSS.

- Divergent Behavior

  1) Getting address and RDNSS from the second router’s RAs, and releasing the IPv6 address and the RDNSS obtained from the DHCPv6 server.

     (More details: when receiving RAs from the first router again, it performs the DHCPv6 Confirm/Reply procedure and gets an IPv6 address and RDNSS from the DHCPv6 server while retaining the ones obtained from the RAs of the second router. Moreover, the RDNSS from router 1 has higher priority than the one from DHCPv6.)

  2) Getting address and RDNSS from the second router’s RAs, and retaining the IPv6 address and the "Domain Search list"

obtained from the DHCPv6 server. (It did not get the RDNSS from the DHCPv6 server, as described in Divergence 2-2.)

(More details: when receiving RAs from the first router again, there is no change; all the obtained information is retained.)

3) Getting address but no RDNSS from the second router’s RAs, and also retaining the IPv6 address and the RDNSS obtained from the DHCPv6 server.

(More details: when receiving RAs from the first router again, there is no change; all the obtained information is retained.)

5. Operational Problems

This section is not a full collection of the potential problems. It is some operational issues that the authors could see at current stage.

5.1. Standalone Stateless DHCPv6 Configuration not available

It is impossible for some hosts to acquire stateless DHCPv6 configuration unless addresses are acquired from either DHCPv6 or SLAAC (Which requires M flag or A flag is TURE).

5.2. Renumbering Issues

According to [RFC6879] a renumbering exercise can include the following steps:

- Causing a host to

  release the SLAAC address and acquire a new address from DHCPv6; or vice-versa.

  release the current SLAAC address and acquire another new SLAAC address (might comes from different source).

  retain current SLAAC or DHCPv6 address and acquire another new address from DHCPv6 or SLAAC.

Ideally, these steps could be initiated by multicasting RA messages onto the link that is being renumbered. Sadly, this is not possible, because the RA messages may elicit a different behavior from each host.

6. Security Considerations

An attacker, without having to install a rogue router, can install a rogue DHCPv6 server and provide IPv6 addresses to Windows 8.1 systems. This can allow her to interact with these systems in a different scope, which, for instance, is not monitored by an IDPS system.

If an attacker wants to perform MiTM (Man in The Middle) using a rogue DNS while legitimates RAs with the O flag set are sent to enforce the use of a DHCPv6 server, the attacker can spoof RAs with the same settings with the legitimate prefix (in order to remain undetectable) but advertising the attacker’s DNS using RDNSS. In this case, Fedora 21, Centos 7 and Ubuntu 14.04 will use the rogue RDNSS (advertised by the RAs) as a first option.

Fedora 21 and Centos 7 behaviour cannot be explored for a MiTM attack using a rogue DNS information either, since the one obtained by the RAs of the first router has a higher priority.

The behaviour of Fedora 21, Centos 7 and Windows 7 can be exploited for DoS purposes. A rogue IPv6 router not only provides its own information to the clients, but it also removes the previous obtained (legitimate) information. The Fedora and Centos behaviour can also be exploited for MiTM purposes by advertising rogue RDNSS by RAs which include RDNSS information.

(Note: the security considerations for specific operating systems are based on the detailed test results as described in Appendix A.)

7. IANA Considerations

This draft does not request any IANA action.

8. Acknowledgements

The authors wish to acknowledge BNRC-BUPT (Broad Network Research Centre in Beijing University of Posts and Telecommunications) for their testing efforts. Special thanks to Xudong Shi, Longyun Yuan and Xiaojian Xue for their extraordinary effort.

Special thanks to Ron Bonica who made a lot of significant contribution to this draft, including draft editing and presentations which dramatically improved this work.

The authors also wish to acknowledge Brian E Carpenter, Ran Atkinson, Mikael Abrahamsson, Tatuya Jinmei, Mark Andrews and Mark Smith for their helpful comments.

9. References

9.1. Normative References

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, DOI 10.17487/RFC2460, December 1998, <http://www.rfc-editor.org/info/rfc2460>.

[RFC4443] Conta, A., Deering, S., and M. Gupta, Ed., "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", RFC 4443, DOI 10.17487/RFC4443, March 2006, <http://www.rfc-editor.org/info/rfc4443>.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, DOI 10.17487/RFC4861, September 2007, <http://www.rfc-editor.org/info/rfc4861>.

[RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, DOI 10.17487/RFC4862, September 2007, <http://www.rfc-editor.org/info/rfc4862>.

[RFC6106] Jeong, J., Park, S., Beloeil, L., and S. Madanapalli, "IPv6 Router Advertisement Options for DNS Configuration", RFC 6106, DOI 10.17487/RFC6106, November 2010, <http://www.rfc-editor.org/info/rfc6106>.

9.2. Informative References

[RFC3315] Droms, R., Ed., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, DOI 10.17487/RFC3315, July 2003, <http://www.rfc-editor.org/info/rfc3315>.

[RFC3736] Droms, R., "Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6", RFC 3736, DOI 10.17487/RFC3736, April 2004, <http://www.rfc-editor.org/info/rfc3736>.

[RFC6879] Jiang, S., Liu, B., and B. Carpenter, "IPv6 Enterprise Network Renumbering Scenarios, Considerations, and Methods", RFC 6879, DOI 10.17487/RFC6879, February 2013, <http://www.rfc-editor.org/info/rfc6879>.

Appendix A. Test Results

The authors from two orgnizations tested different scenarios independent of each other. The following text decribes the two test sets respectively.

A.1. Test Set 1

A.1.1. Test Environment

The test environment was replicated on a single server using VMware. For simplicity of operation, only one host was run at a time. Network elements were as follows:

- Router: Quagga 0.99-19 soft router installed on Ubuntu 11.04 virtual host
- DHCPv6 Server: Dibbler-server installed on Ubuntu 11.04 virtual host
- Host 1: Window 7 / Window 8.1 Virtual Host
- Host 2: Ubuntu 14.04 (Linux Kernel 3.12.0) Virtual Host
- Host 3: Mac OS X v10.9 Virtual Host
- Host 4: IOS 8.0 (model: Apple iPhone 5S, connected via wifi)

A.1.2. Address Auto-configuration Behavior in the Initial State

The bullet list below describes host behavior in the initial state, when the host has not yet acquired any auto-configuration information. Each bullet item represents an input and the behavior elicited by that input.

- A=0, M=0, O=0
  - Windows 8.1 acquired addresses and other information from DHCPv6.
  - All other hosts acquired no configuration information.
- A=0, M=0, O=1
  - Windows 8.1 acquired addresses and other information from DHCPv6.

  - Windows 7, OSX 10.9 and IOS 8.0 acquired other information from DHCPv6.
  - Ubuntu 14.04 acquired no configuration information.

- A=0, M=1, O=0
  - All hosts acquired addresses and other information from DHCPv6.

- A=0, M=1, O=1
  - All hosts acquired addresses and other information from DHCPv6.

- A=1, M=0, O=0
  - Windows 8.1 acquired addresses from SLAAC and DHCPv6. It also acquired non-address information from DHCPv6.
  - All the other host acquired addresses from SLAAC

- A=1, M=0, O=1
  - Windows 8.1 acquired addresses from SLAAC and DHCPv6. It also acquired other information from DHCPv6.
  - All the other hosts acquired addresses from SLAAC and other information from DHCPv6.

- A=1, M=1, O=0
  - All hosts acquired addresses from SLAAC and DHCPv6. They also acquired other information from DHCPv6.

- A=1, M=1, O=1
  - All hosts acquired addresses from SLAAC and DHCPv6. They also acquired other information from DHCPv6.

As showed above, four inputs result in divergent behaviors.

A.1.3. Address Auto-configuration Behavior in State Transitions

The bullet list below describes behavior elicited during state transitions. The value x can represents both 0 and 1.

- Old state (M = x, O = x, A = 1) , New state (M = x, O = x, A = 0) (This means a SLAAC-configured host, which is regardless of DHCPv6 configured or not, receiving A in transition from 1 to 0.)

  - All the hosts retain SLAAC addresses until they expire

- Old state (M = 0, O = x, A = 1), New state (M = 1, O = x, A = 1) (This means a SLAAC-only host receiving M in transition from 0 to 1.)

  - Windows 7 acquires addresses from DHCPv6, immediately.

  - Ubuntu 14.04/OSX 10.9/IOS 8.0 acquires addresses from DHCPv6 only if the SLAAC addresses are allowed to expire

  - Windows 8.1 was not tested because it always acquire addresses from DHCPv6 regardless of the M flag setting.

- Old state (M = 1, O = x, A = x), New state (M = 0, O = x, A = x) (This means a DHCPv6-configured host receiving M in transition from 1 to 0.)

  - Windows 7 immediately released the DHCPv6 address

  - Windows 8.1/Ubuntu 14.04/OSX 10.9/IOS 8.0 keep the DHCPv6 addresses until they expire

- Old state (M = 1, O = x, A = 0), New state (M = 1, O = x, A = 1) (This means a DHCPv6-only host receiving A in transition from 0 to 1.)

  - All host acquire addresses from SLAAC

- Old state (M = 0, O = 1, A = x), New state (M = 1, O = 1, A = x) (This means a Stateless DHCPv6-configured host [RFC3736], which is regardless of SLAAC configured or not, receiving M in transition from 0 to 1 with keeping O=1)

  - Windows 7 acquires addresses and refreshes other information from DHCPv6

  - Ubuntu 14.04/OSX 10.9/IOS 8.0 does nothing

  - Windows 8.1 was not tested because it always acquire addresses from DHCPv6 regardless of the M flag setting.

- Old state (M = 1, O = 1, A = x), New state (M = 0, O = 1, A = x) (This means a Stateful DHCPv6-configured host, which is regardless of SLAAC configured or not, receiving M in transition from 0 to 1 with keeping O=1)

  - Windows 7 released all DHCPv6 addresses and refreshes all DHCPv6 other information.

  - Windows 8.1/Ubuntu 14.04/OSX 10.9/IOS 8.0 does nothing

A.2. Test Set 2

A.2.1. Test Environment

This test was built on real devices. All the devices are located on the same link.

- A DHCPv6 Server and specifically, a DHCP ISC Version 4.3.1 installed in CentOs 6.6. The DHCPv6 server is configured to provide both IPv6 addresses and RDNSS information.

- Two routers Cisco 4321 using Cisco IOS Software version 15.5(1)S.

- The following OS as clients:

  - Fedora 21, kernel version 3.18.3-201 x64

  - Ubuntu 14.04.1 LTS, kernel version 3.13.0-44-generic (rdnssd packet installed)

  - CentOS 7, kernel version 3.10.0-123.13.2.el7

  - Mac OS-X 10.10.2 Yosemite 14.0.0 Darwin

  - Windows 7

  - Windows 8.1

A.2.2. Address/DNS Auto-configuration Behavior of Using Only One IPv6 Router and a DHCPv6 Server

In these scenarios there is two one router and, unless otherwise specified, one DHCPv6 server on the same link. The behaviour of the router and of the DHCPv6 server remain unchanged during the tests.

Case 1: One Router with the Management Flag not Set and a DHCPv6 Server

- Set up

  - One IPv6 Router with M=0, A=1, O=0 and an RDNSS is advertised

  - A DHCPv6 server on the same link advertising IPv6 addresses and RDNSS

- Results

  - Fedora 21, MAC OS-X, CentOS 7 and Ubuntu 14.04 get an IPv6 address and an RDNSS from the IPv6 router only.

  - Windows 7 get an IPv6 address from the router only, but they do not get any DNS information, neither from the router nor from the DHCPv6 server. They also do not get IPv6 address from the DHCPv6 server.

  - Windows 8.1 get an IPv6 address from both the IPv6 router and the DHCPv6 server, despite the fact that the Management flag (M) is not set. They get RDNSS information from the DHCPv6 only.

Case 2: One Router with Conflicting Parameters and a DHCPv6 Server

- Set up

  - One IPv6 Router with M=0, A=1, O=1 and an RDNSS is advertised

  - A DHCPv6 server on the same link advertising IPv6 addresses and RDNSS

- Results

  - Fedora 21, Centos 7 and Ubuntu 14.04 get IPv6 address using SLAAC only (no address from the DHCPv6 server).

    - Fedora 21, Centos 7 get RDNSS from both the RAs and the DHCPv6 server. The RDNSS obtained from the router has a higher priority though.

    - Ubuntu 14.04 gets an RDNSS from the router, and a "domain search list" information from the DHCPv6 server - but not RDNSS information.

  - MAC OS-X also gets RDNSS from both, IPv6 address using SLAAC (no IPv6 address from the DHCPv6 server) but the RDNSS obtained from the DHCPv6 server is first (it has a higher priority). However, the other obtained from the RAs is also present.

  - Windows 7 and Windows 8.1 obtain IPv6 addresses using SLAAC and RDNSS from the DHCPv6 server. They do not get IPv6 address from the DHCPv6 server. Compare the Windows 8.1 behaviour with the previous case.

Case 3: Same as Case 2 but Without a DHCPv6 Server

- Set up
  - One IPv6 Router with M=0, A=1, O=1 and an RDNSS is advertised
  - no DHCPv6 present
- Results
  - Windows 7 and Windows 8.1 get an IPv6 address using SLAAC but they do not get RDNSS information.
  - MAC OS-X, Fedora 21, Centos 7 and Ubuntu 14.04 get an IPv6 address using SLAAC and RDNSS from the RAs.

Case 4: All Flags are Set and a DHCPv6 Server is Present

- Set up
  - One IPv6 Router with M=1, A=1, O=1 and an RDNSS is advertised
  - A DHCPv6 server on the same link advertising IPv6 addresses and RDNSS
- Results
  - Fedora 21 and Centos 7:
    - They get IPv6 address both from SLAAC and DHCPv6 server.
    - They get RDNSS both from RAs and DHCPv6 server.
    - The DNS of the RAs has higher priority.
  - Ubuntu 14.04:
    - It gets IPv6 address both using SLAAC and from the DHCPv6 server.
    - It gets RDNSS from RAs only.
    - From the DHCPv6 server it only gets "Domain Search List" information, no RDNSS.

  - MAC OS-X:
    - It gets IPv6 addresses both using SLAAC and from the DHCPv6 server.
    - It also gets RDNSS both from RAs and the DHCPv6 server.
    - The DNS server of the DHCPv6 has higher priority.
  - Windows 7 and Windows 8.1:
    - They get IPv6 address both from SLAAC and DHCPv6 server.
    - They get RDNSS only from the DHCPv6 server.

Case 5: All Flags are Set and There is No DHCPv6 Server is Present

- Set up
  - One IPv6 Router with M=1, A=1, O=1 and an RDNSS is advertised
  - no DHCPv6 is present
- Results
  - Windows 7 and Windows 8.1 get an IPv6 address using SLAAC but no RDNSS information.
  - MAC OS-X, Fedora 21, Centos 7, Ubuntu 14.04 get an IPv6 address using SLAAC and RDNSS from the RAs.

Case 6: A Prefix is Advertised by RAs but the ’A’ flag is not Set

- Set up
  - An IPv6 Router with M=0, A=0 (while a prefix information is advertised), O=0 and an RDNSS is advertised.
  - DHCPv6 is present
- Results
  - Fedora 21, Centos 7, Ubuntu 14.04 and MAC OS-X:
    - They do not get any IPv6 address (neither from the RAs, nor from the DHCPv6).
    - They get a RDNSS from the router only (not from DHCPv6).

  - Windows 8.1
    - They get IPv6 address and RDNSS from the DHCPv6 server ("last resort" behaviour).
    - They do not get any information (neither IPv6 address not RDNSS) from the router.
  - Windows 7:
    - They get nothing (neither IPv6 address nor RDNSS) from any source (RA or DHCPv6).

A.2.3. Address/DNS Auto-configuration Behavior of Using Two IPv6 Router and a DHCPv6 Server

these scenarios there are two routers on the same link. At first, only one router is present (resembling the "legitimate router)", while the second one joins the link after the clients first configured by the RAs of the first router. Our goal is to examine the behaviour of the clients during the interchange of the RAs from the two different routers.

Case 7: Router 1 Advertising M=0, O=0 and RDNSS, and then Router 2 advertising M=1, O=1 while DHCPv6 is Present

- Set up
  - Initially:
    - One IPv6 router with M=0, O=0, A=1 and RDNSS advertised and 15 seconds time interval of the RAs
  - After a while (when clients are configured by the RAs of the above router):
    - Another IPv6 router with M=1, O=1, no advertised prefix information, and 30 seconds time interval of the RAs.
    - A DHCPv6 server on the same link providing IPv6 addresses and RDNSS.
- Results
  - MAC OS-X and Ubuntu 14.04:
    - Initially they get address and RDNSS from the first router.

    - When they receive RAs from the second router, they never get any information (IPv6 address or RDNSS) from the DHCPv6 server.

  - Windows 7:

    - Initially they get address from the first router - no RDNSS.

    - When they receive RAs from the second router, they never get any information (IPv6 address or RDNSS) from the DHCPv6 server.

  - Fedora 21 and Centos 7:

    - Initially they get IPv6 address and RDNSS from the RAs of the first router. o

    - When they receive an RA from router 2, they also get an IPv6 address and RDNSS from the DHCPv6 server while retaining the ones (IPv6 address and RDNSS) obtained from the RAs of the first router. The RDNSS obtained from the first router has a higher priority than the one obtained from the DHCPv6 server (probably because it was received first). o

    - When they receive again RAs from the first router, they lose/forget the information (IPv6 address and RDNSS) obtained from the DHCPv6 server.

  - Windows 8.1:

    - Initially, they get just an IPv6 address from the first router 1 - no RDNSS information (since they do not implement RFC 6106).

    - When they receive RAs from the second router, then they also get an IPv6 address from the DHCPv6 server, as well as RDNSS from it. They do not lose the IPv6 address obtained by the first router using SLAAC.

    - When they receive RA from the first router, they retain all the obtained so far information (there isn’t any change).

Case 8: (Router 2) Initially M=1, O=1 and DHCPv6, then 2nd Router (Router 1) Rogue RAs Using M=0, O=0 and RDNSS Provided

- Set up

  - Initially:

    - One IPv6 router with M=1, O=1, no advertised prefix information, and 30 seconds time interval of the RAs.
    - A DHCPv6 server on the same link advertising IPv6 addresses and RDNSS.
  - After a while (when clients are configured by the RAs of the above router):
    - Another IPv6 router with M=0, O=0, A=1, RDNSS advertised and 15 seconds time interval of the RAs.

- Results
  - Fedora 21 and Centos 7:
    - At first, they get information (IPv6 address and RDNSS) from the DHCPv6 server.
    - When they receive RAs from the second router, they get address(es) and RDNSS from these RAs. At the same time, the IPv6 address and the RDNSS obtained from the DHCPv6 server are gone.
    - When they receives again an RA from the first router, they perform the DHCPv6 Confirm/Reply procedure and they get an IPv6 address and RDNSS from the DHCPv6 server while retaining the ones obtained from the RAs of the second router. Moreover, the RDNSS from router 1 has higher priority than the one from DHCPv6.
  - Ubuntu 14.04:
    - At first, it gets information (IPv6 address and RDNSS) from the DHCPv6 server.
    - When it receives RAs from the second router, it also gets information from it, but it does not lose the information obtained from the DHCPv6 server. It retains both. It only gets "Domain Search list" from the DHCPv6 server-no RDNSS information.
    - When it receives RAs from the first router, there is no change; it retains all the obtained information.
  - Windows 7:

    - Initially they get IPv6 address and RDNSS from the DHCPv6 server.
    - When they get RAs from the second router, they lose this information (IPv6 address and RDNSS obtained from the DHCPv6 server) and they get only SLAAC addresses using the RAs of the second router-no RDNSS.
    - When they receive RAs from the first router again, they get RDNSS and IPv6 address from the DHCPv6 server, but they also keep the SLAAC addresses.

- Windows 8.1:
    - Initially they get information (IPv6 address and RDNSS) from the DHCPv6 server.
    - When they receive RAs from the second router, they never get any information from them.

- MAC OS-X:
    - Initially it gets information (IPv6 address and RDNSS) from the DHCPv6 server.
    - When it gets RAs from the second router, it also gets a SLAAC IPv6 address but no RDNSS information from the RAs of this router. It also does not lose any information obtained from DHCPv6.
    - When it gets RAs from the first router again, the situation does not change (IPv6 addresses from both the DHCPv6 and SLAAC process are retained, but RDNSS information only from the DHCPv6 server).

Authors' Addresses

Bing Liu
Huawei Technologies
Q14, Huawei Campus, No.156 Beiqing Road
Hai-Dian District, Beijing, 100095
P.R. China

Email: leo.liubing@huawei.com

Sheng Jiang
Huawei Technologies
Q14, Huawei Campus, No.156 Beiqing Road
Hai-Dian District, Beijing, 100095
P.R. China

Email: jiangsheng@huawei.com

Xiangyang Gong
BUPT University
No.3 Teaching Building
Beijing University of Posts and Telecommunications (BUPT)
No.10 Xi-Tu-Cheng Rd.
Hai-Dian District, Beijing
P.R. China

Email: xygong@bupt.edu.cn

Wendong Wang
BUPT University
No.3 Teaching Building
Beijing University of Posts and Telecommunications (BUPT)
No.10 Xi-Tu-Cheng Rd.
Hai-Dian District, Beijing
P.R. China

Email: wdwang@bupt.edu.cn

Enno Rey
ERNW GmbH

Email: erey@ernw.de

Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 22, 2015

G. Chen
H. Deng
China Mobile
D. Michaud
Rogers Communications
J. Korhonen
Broadcom
M. Boucadair
France Telecom
A. Vizdal
Deutsche Telekom AG

October 19, 2014

Analysis of Failure Cases in IPv6 Roaming Scenarios
draft-ietf-v6ops-ipv6-roaming-analysis-07

Abstract

This document identifies a set of failure cases that may be encountered by IPv6-enabled mobile customers in roaming scenarios. The analysis reveals that the failure causes include improper configurations, incomplete functionality support in equipment, and inconsistent IPv6 deployment strategies between the home and the visited networks.

Status of This Memo

This Internet-Draft will expire on April 22, 2015.

Copyright Notice

Table of Contents

1. Introduction

Many Mobile Operators have deployed IPv6, or are about to, in their operational networks. A customer in such a network can be provided IPv6 connectivity if their User Equipment (UE) is IPv6-compliant. Operators may adopt various approaches to deploy IPv6 in mobile networks such as the solutions described in [TR23.975]). Depending on network conditions, either dual-stack or IPv6-only deployment schemes can be enabled.

A detailed overview of IPv6 support in 3GPP architectures is provided in [RFC6459].

It has been observed and reported that a mobile subscriber roaming around a different operator’s areas may experience service disruption due to inconsistent configurations and incomplete functionality of equipment in the network. This document focuses on these issues.

1.1. Terminology

This document makes use of these terms:

- Mobile networks refer to 3GPP mobile networks.
- Mobile UE denotes a 3GPP device which can be connected to 3GPP mobile networks.
- The Public Land Mobile Network (PLMN) is a network that is operated by a single administrative entity. A PLMN (and therefore also an operator) is identified by the Mobile Country Code (MCC) and the Mobile Network Code (MNC). Each (telecommunications) operator providing mobile services has its own PLMN [RFC6459].
- The Home Location Register (HLR) is a pre-Release-5 database (but is also used in Release-5 and later networks in real deployments) that contains subscriber data and information related to call routing. All subscribers of an operator and the subscribers’ enabled services are provisioned in the HLR [RFC6459].
- The Home Subscriber Server (HSS) is a database for a given subscriber and was introduced in 3GPP Release-5. It is the entity containing the subscription-related information to support the network entities actually handling calls/sessions [RFC6459].

"HLR/HSS" is used collectively for the subscriber database unless referring to the failure case related to General Packet Radio Service (GPRS) Subscriber data from the HLR.

An overview of key 3GPP functional elements is documented in [RFC6459].

"Mobile device" and "mobile UE" are used interchangeably.

2. Background

2.1. Roaming Architecture: An Overview

Roaming occurs in two scenarios:

- International roaming: a mobile UE enters a visited network operated by a different operator, where a different Public Land Mobile Network (PLMN) code is used. The UEs could, either in an automatic mode or in a manual mode, attach to the visited PLMN.

- Intra-PLMN mobility: an operator may have one or multiple PLMN codes. A mobile UE could pre-configure the codes to identify the Home PLMN (HPLMN) or Equivalent HPLMN (EHPLMN). Intra-PLMN mobility allows the UE moving to a different area of HPLMN and EHPLMN. When the subscriber profile is not stored in the visited area, HLR/HSS in the Home area will transmit the profile to Serving GPRS Support Node (SGSN)/Mobility Management Entity (MME) in the visited area so as to complete network attachment.

When a UE is turned on or is transferred via a hand-over to a visited network, the mobile device will scan all radio channels and find available PLMNs to attach to. The SGSN or the MME in the visited networks must contact the HLR or HSS to retrieve the subscriber profile.

Steering of roaming may also be used by the HPLMN to further restrict which of the available networks the UE may be attached to. Once the authentication and registration stage is completed, the Packet Data Protocol (PDP) or Packet Data Networks (PDN) activation and traffic flows may be operated differently according to the subscriber profile stored in the HLR or the HSS.

The following sub-sections describe two roaming modes: Home routed traffic (Section 2.1.1) and Local breakout (Section 2.1.2).

2.1.1. Home Routed Mode

In this mode, the subscriber’s UE gets IP addresses from the home network. All traffic belonging to that UE is therefore routed to the home network (Figure 1).

GPRS roaming exchange (GRX) or Internetwork Packet Exchange (IPX) networks [IR.34] are likely to be invoked as the transit network to deliver the traffic. This is the main mode for international roaming of Internet data services to facilitate the charging process between the two involved operators.

```
 +-----------------------------+            +------------------------+
 |Visited Network              |            |Home Network            |
 |  +----+        +--------+   | (GRX/IPX)  |    +--------+ Traffic Flow
 |  | UE |=======>|SGSN/MME|====================>|GGSN/PGW|============>
 |  +----+        +--------+   | Signaling  |    +--------+          |
 |                     |------------------------>+--------+          |
 |                             |            |    |HLR/HSS |          |
 |                             |            |    +--------+          |
 +-----------------------------+            +------------------------+
```

Figure 1: Home Routed Traffic

2.1.2. Local Breakout Mode

In the local breakout mode, IP addresses are assigned by the visited network to a roaming mobile UE. Unlike the home mode, the traffic doesn’t have to traverse GRX/IPX; it is offloaded locally at a network node close to that device’s point of attachment in the visited network. This mode ensures a more optimized forwarding path for the delivery of packets belonging to a visiting UE (Figure 2).

```
   +----------------------------+            +----------------+
   |Visited Network             |            |Home Network    |
   |  +----+        +--------+  | Signaling  |    +--------+  |
   |  | UE |=======>|SGSN/MME|------------------->|HLR/HSS |  |
   |  +----+        +--------+  | (GRX/IPX)  |    +--------+  |
   |                    ||      |            |                |
   |                +--------+  |            |                |
   |                |GGSN/PGW|  |            |                |
   |                +--------+  |            |                |
   |      Traffic Flow  ||      |            |                |
   +--------------------||------+            +----------------+
                        \/
```

Figure 2: Local Breakout

The international roaming of IP Multimedia Subsystem (IMS) based services, e.g., Voice over LTE (VoLTE)[IR.92], is claimed to select the local breakout mode in [IR.65]. Data service roaming across different areas within an operator network might use local breakout mode in order to get more efficient traffic forwarding and also ease emergency services. The local breakout mode could also be applied to an operator’s alliance for international roaming of data service.

EU Roaming Regulation III [EU-Roaming-III] involves local breakout mode allowing European subscribers roaming in European 2G/3G networks to have their Internet data routed directly to the Internet from their current VPLMN.

Specific local breakout-related configuration considerations are listed below:

- Operators may add the APN-OI-Replacement flag defined in 3GPP [TS29.272] into the user’s subscription-data. The visited network indicates a local domain name to replace the user requested Access Point Name (APN). Consequently, the traffic would be steered to the visited network. Those functions are normally deployed for the intra-PLMN mobility cases.

- Operators may also configure the VPLMN-Dynamic-Address-Allowed flag [TS29.272] in the user’s profile to enable local breakout mode in Visited Public Land Mobile Networks (VPLMNs).

- 3GPP specified Selected IP Traffic Offload (SIPTO) function [TS23.401] since Release 10 in order to get efficient route paths. It enables an operator to offload a portion of the traffic at a network node close to the visiting UE’s point of attachment to the visited network.

- GSMA has defined Roaming Architecture for Voice over LTE with Local Breakout (RAVEL) [IR.65] as the IMS international roaming architecture. Local breakout mode has been adopted for the IMS roaming architecture.

2.2. Typical Roaming Scenarios

Three stages occur when a subscriber roams to a visited network and intends to invoke services:

- Network attachment: this occurs when the UE enters a visited network. During the attachment phase, the visited network should authenticate the subscriber and make a location update to the HSS/ HLR in the home network of the subscriber. Accordingly, the subscriber profile is offered from the HSS/HLR. The subscriber profile contains the allowed Access Point Names (APN), the allowed PDP/PDN Types and rules regarding the routing of data sessions (i.e., home routed or local breakout mode) [TS29.272]. The SGSN/ MME in the visited network can use this information to facilitate the subsequent PDP/PDN session creation.

- PDP/PDN context creation: this occurs after the subscriber UE has been successfully attached to the network. This stage is integrated with the attachment stage in the case of 4G, but is a separate process in 2/3G. 3GPP specifies three types of PDP/PDN to describe connections, i.e., PDP/PDN Type IPv4, PDP/PDN Type IPv6 and PDP/ PDN Type IPv4v6. When a subscriber creates a data session, their device requests a particular PDP/PDN Type. The allowed PDP/PDN types for that subscriber are learned in the attachment stage. Hence, SGSN/MME could initiate PDP/PDN request to GGSN/PGW modulo subscription grants.

- Service requests: when the PDP/PDN context is created successfully, UEs may launch applications and request services based on the allocated IP addresses. The service traffic will be transmitted via the visited network.

Failures that occur at the attachment stage (Section 3) are independent of home routed and the local breakout mode. Most failure cases in the PDP/PDN context creation (Section 4) and service requests (Section 5) occur in the local breakout mode.

3. Failure Case in the Network Attachment

3GPP specified PDP/PDN type IPv4v6 in order to allow a UE get both an IPv4 address and an IPv6 prefix within a single PDP/PDN bearer. This option is stored as a part of subscription data for a subscriber in the HLR/HSS. PDP/PDN type IPv4v6 has been introduced at the inception of Evolved Packet System (EPS) in 4G networks.

The nodes in 4G networks should present no issues with the handling of this PDN type. However, the level of support varies in 2/3G networks depending on SGSN software version. In theory, S4-SGSN (i.e., an SGSN with S4 interface) supports the PDP/PDN type IPv4v6 since Release 8 and a Gn-SGSN (i.e., the SGSN with Gn interface) supports it since Release 9. In most cases, operators normally use Gn-SGSN to connect either GGSN in 3G or Packet Data Network Gateway (PGW) in 4G.

The MAP (Mobile Application Part) protocol, as defined in 3GPP [TS29.002], is used over the Gr interface between SGSN and HLR. The MAP Information Element (IE) "ext-pdp-Type" contains the IPv4v6 PDP Type that is conveyed to SGSN from the HLR within the Insert Subscriber Data (ISD) MAP operation. If the SGSN does not support the IPv4v6 PDP Type, it will not support the "ext-pdp-Type" IE and consequently it must silently discard that IE and continue processing of the rest of the ISD MAP message. An issue that has been observed is that multiple SGSNs are unable to correctly process a subscriber’s data received in the Insert Subscriber Data Procedure [TS23.060]. As a consequence, it will likely discard the subscriber attach request. This is erroneous behavior due to the equipment not being compliant with 3GPP Release 9.

In order to avoid encountering this attach problem at a visited SGSN, both operators should make a comprehensive roaming agreement to support IPv6 and ensure that it aligns with the GSMA documents, e.g., [IR.33], [IR.88] and [IR.21]. Such an agreement requires the visited operator to get the necessary patch on all its SGSN nodes to support the "ext-pdp-Type" MAP IE sent by the HLR. To ensure data session continuity in Radio Access Technology (RAT) handovers the PDN Type sent by the HSS to the MME could be consistent with the PDP Type sent by the HLR to the Gn-SGSN. Where roaming agreements and visited SGSN nodes have not been updated, the HPLMN also has to make use of specific implementations (not standardized by 3GPP, discussed further in Section 6) in the HLR/HSS of the home network. That is, when the HLR/HSS receives an Update Location message from a visited SGSN not known to support dual-stack in a single bearer, subscription data allowing only PDP/PDN type IPv4 or IPv6 will be sent to that SGSN in the Insert Subscriber Data procedure. This guarantees that the user profile is compatible with the visited SGSN/MME capability. In addition, HSS may not have to change, if the PGW is aware of subscriber’s roaming status and only restricts the accepted PDN type consistent with PDP type sent by the HLR. For example, an AAA server may coordinate with the PGW to decide the allowed PDN type.

Alternatively, HPLMNs without the non-standardized capability to suppress the sending of "ext-pdp-Type" by the HLR may have to remove this attribute from APNs with roaming service. PDN Type IPv4v6 must also be removed from the corresponding profile for the APN in the HSS. This will restrict their roaming UEs to only IPv4 or IPv6 PDP/PDN activation. This alternative has problems:

- The HPLMN cannot support dual-stack in a single bearer at home either where the APN profile in the HLR/HSS is also used for roaming.
- The UE may set-up separate parallel bearers for IPv4 and IPv6 where only single stack IPv4 or IPv6 service is preferred by the operator.

4. Failure Cases in the PDP/PDN Creation

When a subscriber’s UE succeeds in the attach stage, the IP allocation process takes place to retrieve IP addresses. In general, a PDP/PDN type IPv4v6 request implicitly allows the network side to make several IP assignment options, including IPv4-only, IPv6-only, IPv4 and IPv6 in single PDP/PDN bearer, IPv4 and IPv6 in separated PDP/PDN bearers.

A PDP/PDN type IPv4 or IPv6 restricts the network side to only allocate requested IP address family.

This section summarizes several failures in the Home Routed (HR) and Local Breakout (LBO) mode as shown in Table 1.

Case#	UE request	PDP/PDN IP Type permitted on GGSN/PGW	Mode
#1	IPv4v6	IPv4v6	HR
	IPv4v6	IPv4 or IPv6	LBO
#2	IPv6	IPv6	HR
#3	IPv4	IPv6	HR
#4	IPv6	IPv4	LBO

Table 1: Failure Cases in the PDP/PDN Creation

4.1. Case 1: Splitting Dual-stack Bearer

Dual-stack capability is provided using separate PDP/PDN activation in the visited network that doesn’t support PDP/PDN type IPv4v6. That means only separate parallel single-stack IPv4 and IPv6 PDP/PDN connections are allowed to be initiated to separately allocate an IPv4 address and an IPv6 prefix. The SGSN does not support the Dual Address Bearer Flag (DAF) or does not set DAF because the operator uses single addressing per bearer to support interworking with nodes of earlier releases. Regardless of home routed or local breakout mode, GGSN/PGW will change PDN/PDP type to a single address PDP/PDN type and return the Session Management (SM) Cause #52 "Single address bearers only allowed" or SM Cause #28 "Unknown PDP address or PDP type" as per [TS24.008] and [TS24.301] to the UE. In this case, the UE may make another PDP/PDN request with a single address PDP type (IPv4 or IPv6) other than the one already activated.

This approach suffers from the followings drawbacks:

- The parallel PDP/PDN activation would likely double PDP/PDN bearer resource on the network side and Radio Access Bearer (RAB) resource on the RAN side. It also impacts the capacity of the GGSN/PGW, since only a certain amount of PDP/PDN activation is allowed on those nodes.

- Some networks may only allow one PDP/PDN be alive for each subscriber. For example, an IPv6 PDP/PDN will be rejected if the subscriber has an active IPv4 PDP/PDN. Therefore, the subscriber would not be able to obtain the IPv6 connection in the visited network. It is even worse as they may have a risk of losing all data connectivity if the IPv6 PDP gets rejected with a permanent error at the APN-level and not an error specific to the PDP-Type IPv6 requested.

- Additional correlations between those two PDP/PDN contexts are required on the charging system.
- Policy and Charging Rules Function (PCRF) [TS29.212]/ Policy and Charging Enforcement Function (PCEF) treats the IPv4 and IPv6 session as independent and performs different Quality of Service (QoS) policies. The subscriber may have unstable experiences due to different behaviors on each IP version connection.
- Mobile devices may have a limitation on allowed simultaneous PDP/ PDN contexts. Excessive PDP/PDN activation may result in service disruption.

In order to avoid the issue, the roaming agreement in the home routed mode should make sure the visited SGSN supports and set the DAF. Since the PDP/PDN type IPv4v6 is supported in the GGSN/PGW of home network, it’s expected that the visited SGSN/MME could create dual-stack bearer as UE requested.

In the local breakout mode, the visited SGSN may only allow single IP version addressing. In this case, DAF on visited SGSN/MME has to be unset. One approach is to set a dedicated Access Point Name (APN) [TS23.003] profile to only request PDP/PDN type IPv4 in the roaming network. Some operators may also consider not adopting the local breakout mode to avoid the risks.

4.2. Case 2: IPv6 PDP/PDN Unsupported

PDP/PDN type IPv6 has good compatibility to visited networks during the network attachment. In order to support the IPv6-only visitors, SGSN/MME in the visited network is required to accept IPv6-only PDP/ PDN activation requests and enable IPv6 on user plane towards the home network.

In some cases, IPv6-only visitors may still be subject to the SGSN capability in visited networks. This becomes especially risky if the home operator performs roaming steering targeted to an operator that doesn’t allow IPv6. The visited SGSN may just directly reject the PDP context activation. Therefore, it’s expected that visited network is IPv6 roaming-friendly to enable the functions on SGSN/MME by default. Otherwise, operators may consider steering the roaming traffic to the IPv6-enable visited network that has IPv6 roaming agreement.

4.3. Case 3: Inappropriate Roaming APN Set

If IPv6 single stack with the home routed mode is deployed, the requested PDP/PDN type should also be IPv6. Some implementations that support roaming APN profile may set IPv4 as the default PDP/PDN type, since the visited network is incapable of supporting PDP/PDN types IPv4v6 (Section 4.1) and IPv6 (Section 4.2). The PDP/PDN request will fail because the APN in the home network only allows IPv6. Therefore, the roaming APN have to be compliant with the home network configuration when home routed mode is adopted.

4.4. Case 4: Fallback Failure

In the local breakout mode, PDP/PDN type IPv6 should have no issues to pass through network attachment process, since 3GPP specified the PDP/PDN type IPv6 as early as PDP/PDN type IPv4. When a visitor requests PDP/PDN type IPv6, the network should only return the expected IPv6 prefix. The UE may fail to get an IPv6 prefix if the visited network only allocates an IPv4 address. In this case, the visited network will reject the request and send the cause code to the UE.

A proper fallback scheme for PDP/PDN type IPv6 is desirable, however there is no standard way to specify this behavior. Roaming APN profile could help to address the issue by setting PDP/PDN type IPv4. For instance, the Android system solves the issue by configuring the roaming protocol to IPv4 for the Access Point Name (APN). It guarantees that UE will always initiate a PDP/PDN type IPv4 in the roaming area.

5. Failure Cases in the Service Requests

After the successful network attachment and IP address allocation, applications could start to request service based on the activated PDP/PDN context. The service request may depend on specific IP family or network collaboration. If traffic is offloaded locally (Section 2.1.2), the visited network may not be able to accommodate UE’s service requests. This section describes the failures.

5.1. Lack of IPv6 Support in Applications

Operators may only allow IPv6 in the IMS APN. VoLTE [IR.92] or Rich Communication Suite (RCS) [RCC.07] use the APN to offer the voice service for visitors. The IMS roaming in RAVEL architecture [IR.65] offloads voice and video traffic in the visited network, therefore a dual-stack visitor can only be assigned with an IPv6 prefix but no IPv4 address. If the applications can’t support IPv6, the service is likely to fail.

Translation-based methods, for example 464xlat [RFC6877] or Bump-in-the-host (BIH) [RFC6535], may help to address the issue if there are IPv6 compatibility problems. The translation function could be enabled in an IPv6-only network and disabled in a dual-stack or IPv4 network, therefore the IPv4 applications only get the translation in the IPv6 network and perform normally in an IPv4 or dual-stack network.

5.2. 464xlat Support

464xlat[RFC6877] is proposed to address the IPv4 compatibility issue in an IPv6-only connectivity environment. The customer-side translator (CLAT) function on a mobile device is likely used in conjunction with a PDP/PDN IPv6 type request and cooperates with a remote NAT64 [RFC6146] device.

464xlat may use the mechanism defined in [RFC7050] or [RFC7225] to detect the presence of NAT64 devices and to learn the IPv6 prefix used for protocol translation[RFC6052].

In the local breakout approach, when a UE with the 464xlat function roaming on an IPv6 visited network may encounter various situations. For example, the visited network may not deploy DNS64 [RFC6147] but only NAT64, CLAT may not be able to discover the provider-side translator (PLAT) translation IPv6 prefix used as a destination of the PLAT. If the visited network doesn’t deploy NAT64 and DNS64, 464xlat can’t perform successfully due to the lack of PLAT collaboration. Even in the case of the presence of NAT64 and DNS64, pre-configured PLAT-side IPv6 prefix in the CLAT may cause the failure because it can’t match the PLAT translation.

Considering the various network’s situations, operators may turn off local breakout and use the home routed mode to perform 464xlat. Alternatively, UE may support the different roaming profile configurations to adopt 464xlat in the home networks and use IPv4-only in the visited networks.

6. HLR/HSS User Profile Setting

A proper user profile configuration would provide a deterministic outcome to the PDP/PDN creation stage where dual-stack, IPv4-only and IPv6-only connectivity requests may come from devices. The HLR/HSS may have to apply extra logic (not standardized by 3GPP) to achieve this. It is also desirable that the network could set-up connectivity of any requested PDP/PDN context type.

The following are examples to illustrate the settings for the scenarios and decision criteria to apply when returning user profile information to the visited SGSN.

```
user profile #1:

PDP-Context ::= SEQUENCE {
pdp-ContextId ContextId,
pdp-Type  PDP-Type-IPv4
  ....
ext-pdp-Type PDP-Type-IPv4v6
  ...
}


user profile #2:

PDP-Context ::= SEQUENCE {
pdp-ContextId ContextId,
pdp-Type  PDP-Type-IPv6
  ....
}
```

Scenario 1: Support of IPv6-only, IPv4-only and dual-stack devices.

The full PDP-context parameters are referred to Section 17.7.1 "Mobile Service date types" of [TS29.002]. User profiles #1 and #2 share the same "ContextId". The setting of user profile #1 enables IPv4-only and dual-stack devices to work. And, the user profile #2 fulfills the request if the device asks for IPv6 only PDP context.

```
user profile #1:

PDP-Context ::= SEQUENCE {
pdp-ContextId ContextId,
pdp-Type  PDP-Type-IPv4
  ....
ext-pdp-Type PDP-Type-IPv4v6
  ...
}


user profile #2:

PDP-Context ::= SEQUENCE {
pdp-ContextId ContextId,
pdp-Type  PDP-Type-IPv4
  ....
}
```

Scenario 2: Support of dual-stack devices with pre-R9 vSGSN access.

User profiles #1 and #2 share the same "ContextId". If a visited SGSN is identified as early as pre-Release 9, the HLR/HSS should only send user profile#2 to the visited SGSN.

7. Discussion

Several failure cases have been discussed in this document. It has been illustrated that the major problems happen at three stages, i.e., the initial network attachment, the PDP/PDN creation and service requests.

In the network attachment stage, PDP/PDN type IPv4v6 is the major concern to the visited pre-Release 9 SGSN. 3GPP didn’t specify PDP/PDN type IPv4v6 in the earlier releases. That PDP/PDN type is supported in new-built EPS network, but isn’t supported well in the third generation network. Visited SGSNs may discard the subscriber’s attach requests because the SGSN is unable to correctly process PDP/PDN type IPv4v6. Operators may have to adopt temporary solutions unless all the interworking nodes (i.e., the SGSN) in the visited network have been upgraded to support the ext-PDP-Type feature.

In the PDP/PDN creation stage, PDP/PDN types IPv4v6 and IPv6 support on the visited SGSN is the major concern. It has been observed that IPv6 single stack with the home routed mode is a viable approach to deploy IPv6. It is desirable that the visited SGSN could enable IPv6 on the user plane by default. For support of the PDP/PDN type IPv4v6, it is suggested to set the DAF. As a complementary function, the implementation of roaming APN configuration is useful to accommodate the visited network. However, it should consider roaming architecture and permitted PDP/PDN type to make proper setting on the UE. Roaming APN in the home routed mode is recommended to align with home network profile setting. In the local breakout case, PDP/PDN type IPv4 could be selected as a safe way to initiate PDP/PDN activation.

In the service requests stage, the failure cases mostly occur in the local breakout case. The visited network may not be able to satisfy the requested capability from applications or UEs. Operators may consider using home routed mode to avoid these problems. Several solutions either in the network side or mobile device side can also help to address the issue. For example,

- 464xlat could help IPv4 applications access IPv6 visited networks.

- Networks can deploy an AAA server to coordinate the mobile device capability. Once the GGSN/PGW receives the session creation request, it will initiate an Access-Request to an AAA server in the home network via the RADIUS protocol. The Access-Request contains subscriber and visited network information, e.g., PDP/PDN Type, International Mobile Equipment Id (IMEI), Software Version (SV) and visited SGSN/MME location code, etc. The AAA server could take mobile device capability and combine it with the visited network information to ultimately determine the type of session to be created, i.e., IPv4, IPv6 or IPv4v6.

8. IANA Considerations

This document makes no request of IANA.

9. Security Considerations

Although this document defines neither a new architecture nor a new protocol, the reader is encouraged to refer to [RFC6459] for a generic discussion on IPv6-related security considerations.

10. Acknowledgements

Many thanks to F. Baker and J. Brzozowski for their support.

This document is the result of the IETF v6ops IPv6-Roaming design team effort.

The authors would like to thank Mikael Abrahamsson, Victor Kuarsingh, Heatley Nick, Alexandru Petrescu, Tore Anderson, Cameron Byrne, Holger Metschulat and Geir Egeland for their helpful discussions and comments.

The authors especially thank Fred Baker and Ross Chandler for their efforts and contributions which substantially improved the readability of the document.

11. References

11.1. Normative References

[IR.21] Global System for Mobile Communications Association, GSMA., "Roaming Database, Structure and Updating Procedures", July 2012.

[IR.65] Global System for Mobile Communications Association, GSMA., "IMS Roaming & Interworking Guidelines", May 2012.

[RFC6146] Bagnulo, M., Matthews, P., and I. van Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", RFC 6146, April 2011.

[RFC6147] Bagnulo, M., Sullivan, A., Matthews, P., and I. van Beijnum, "DNS64: DNS Extensions for Network Address Translation from IPv6 Clients to IPv4 Servers", RFC 6147, April 2011.

[RFC6877] Mawatari, M., Kawashima, M., and C. Byrne, "464XLAT: Combination of Stateful and Stateless Translation", RFC 6877, April 2013.

[TS23.060] 3rd Generation Partnership Project, 3GPP., "General Packet Radio Service (GPRS); Service description; Stage 2 v9.00", March 2009.

[TS23.401] 3rd Generation Partnership Project, 3GPP., "General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access v9.00", March 2009.

[TS29.002] 3rd Generation Partnership Project, 3GPP., "Mobile Application Part (MAP) specification v9.12.0", December 2009.

[TS29.272]
3rd Generation Partnership Project, 3GPP., "Mobility Management Entity (MME) and Serving GPRS Support Node (SGSN) related interfaces based on Diameter protocol v9.00", September 2009.

11.2. Informative References

[EU-Roaming-III]
"http://www.amdocs.com/Products/Revenue-Management/Documents/amdocs-eu-roaming-regulation-III-solution.pdf", July 2013.

[IR.33] Global System for Mobile Communications Association, GSMA., "GPRS Roaming Guidelines", July 2012.

[IR.34] Global System for Mobile Communications Association, GSMA., "Guidelines for IPX Provider networks", November 2013.

[IR.88] Global System for Mobile Communications Association, GSMA., "LTE Roaming Guidelines", January 2012.

[IR.92] Global System for Mobile Communications Association (GSMA), , "IMS Profile for Voice and SMS Version 7.0", March 2013.

[RCC.07] Global System for Mobile Communications Association (GSMA), , "Rich Communication Suite 5.1 Advanced Communications Services and Client Specification Version 4.0", November 2013.

[RFC6052] Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", RFC 6052, October 2010.

[RFC6459] Korhonen, J., Soininen, J., Patil, B., Savolainen, T., Bajko, G., and K. Iisakkila, "IPv6 in 3rd Generation Partnership Project (3GPP) Evolved Packet System (EPS)", RFC 6459, January 2012.

[RFC6535] Huang, B., Deng, H., and T. Savolainen, "Dual-Stack Hosts Using "Bump-in-the-Host" (BIH)", RFC 6535, February 2012.

[RFC7050] Savolainen, T., Korhonen, J., and D. Wing, "Discovery of the IPv6 Prefix Used for IPv6 Address Synthesis", RFC 7050, November 2013.

[RFC7225] Boucadair, M., "Discovering NAT64 IPv6 Prefixes Using the Port Control Protocol (PCP)", RFC 7225, May 2014.

[TR23.975] 3rd Generation Partnership Project, 3GPP., "IPv6 migration guidelines", June 2011.

[TS23.003] 3rd Generation Partnership Project, 3GPP., "Numbering, addressing and identification v9.0.0", September 2009.

[TS24.008] 3rd Generation Partnership Project, 3GPP., "Mobile radio interface Layer 3 specification; Core network protocols; Stage 3 v9.00", September 2009.

[TS24.301] 3rd Generation Partnership Project, 3GPP., "Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS) ; Stage 3 v9.00", September 2009.

[TS29.212] 3rd Generation Partnership Project, 3GPP., "Policy and Charging Control (PCC); Reference points v9.0.0", September 2009.

Authors’ Addresses

Gang Chen
China Mobile
53A,Xibianmennei Ave.,
Xuanwu District,
Beijing 100053
China

Email: phdgang@gmail.com

Hui Deng
China Mobile
53A,Xibianmennei Ave.,
Xuanwu District,
Beijing 100053
China

Email: denghui@chinamobile.com

Dave Michaud
Rogers Communications
8200 Dixie Rd.
Brampton, ON L6T 0C1
Canada

Email: dave.michaud@rci.rogers.com

Jouni Korhonen
Broadcom
Porkkalankatu 24
FIN-00180 Helsinki, Finland

Email: jouni.nospam@gmail.com

Mohamed Boucadair
France Telecom
Rennes,
35000
France

Email: mohamed.boucadair@orange.com

Vizdal Ales
Deutsche Telekom AG
Tomickova 2144/1
Prague 4, 149 00
Czech Republic

Email: ales.vizdal@t-mobile.cz

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: November 3, 2015

B. Liu
S. Jiang
Huawei Technologies

May 2, 2015

Considerations For Using Unique Local Addresses
draft-ietf-v6ops-ula-usage-recommendations-05

Abstract

This document provides considerations for using IPv6 Unique Local Addresses (ULAs). It identifies cases where ULA addresses are helpful as well as potential problems that their use could introduce, based on an analysis of different ULA usage scenarios.

Status of This Memo

This Internet-Draft will expire on November 3, 2015.

Copyright Notice

Table of Contents

1. Introduction

Unique Local Addresses (ULAs) are defined in [RFC4193] as provider-independent prefixes that can be used locally, for example, on isolated networks, internal networks, or VPNs. Although ULAs may be treated like addresses of global scope by applications, normally they are not used on the public Internet. ULAs are a possible alternative to site-local addresses (deprecated in [RFC3879]) in some situations, but there are differences between the two address types.

The use of ULAs in various types of networks has been confusing to network operators. This document aims to clarify the advantages and disadvantages of ULAs and how they can be most appropriately used.

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119] when they appear in ALL CAPS. When these words are not in ALL CAPS (such as "should" or "Should"), they have their usual English meanings, and are not to be interpreted as [RFC2119] key words.

3. Analysis of ULA Features

3.1. Automatically Generated

ULA prefixes can be automatically generated using the algorithms described in [RFC4193]. This feature allows automatic prefix allocation. Thus one can get a network working immediately without applying for prefix(es) from an RIR/LIR (Regional Internet Registry/ Local Internet Registry).

3.2. Globally Unique

ULAs are intended to have an extremely low probability of collision. Since multiple networks in which the hosts have been assigned with ULAs may occasionally be merged into one network, this uniqueness is necessary. The randomization of 40 bits in a ULA prefix is considered sufficient enough to ensure a high degree of uniqueness (refer to [RFC4193] Section 3.2.3 for details) and simplifies merging of networks by avoiding the need to renumber overlapping IP address space. Such overlapping was a major drawback to the deployment of private [RFC1918] addresses in IPv4.

Note that, as described in [RFC4864], applications may treat ULAs in practice like global-scope addresses, but address selection algorithms may need to distinguish between ULAs and Global-scope Unicast Addresses (GUAs) to ensure bidirectional communications. As a further note, the default address selection policy table in [RFC6724]) responds to this requirement.

3.3. Independent Address Space

ULAs provide internal address independence in IPv6 since they can be used for internal communications even without Internet connectivity. They need no registration, so they can support on-demand usage and do not carry any RIR/LIR burden of documentation or fees.

3.4. Well Known Prefix

The prefixes of ULAs are well known thus they are easily identified and filtered.

This feature is convenient for management of security policies and troubleshooting. For example, network administrators can segregate packets containing data which must stay in the internal network by assigning ULAs to internal servers. Externally-destined data can be sent to the Internet or telecommunication network by a separate function, through an appropriate gateway/firewall.

3.5. Stable or Temporary Prefix

A ULA prefix can be generated once, at installation time or factory reset, and then possibly never be changed. Alternatively, it can be regenerated regularly, depending on deployment requirements.

4. Analysis and Operational Considerations of Scenarios Using ULAs

4.1. Isolated Networks

IP is used ubiquitously. Some networks like industrial control bus (e.g. [RS-485], [SCADA], or even non-networked digital interfaces like [MIL-STD-1397] have begun to use IP. In these kinds of networks, the system may lack the ability to communicate with the public networks.

As another example, there may be some networks in which the equipment has the technical capability to connect to the Internet, but is prohibited by administration or just temporarily not connected. These networks may include separate financial networks, lab networks. machine-to-machine (e.g. vehicle networks), sensor networks, or even normal LANs, and can include very large numbers of addresses.

Serious disadvantages and impact on applications due to the use of ambiguous address space have been well documented in [RFC1918]. However, ULA is a straightforward way to assign the IP addresses in the kinds of networks just described, with minimal administrative cost or burden. Also, ULAs fit in multiple subnet scenarios, in which each subnet has its own ULA prefix. For example, when we assign vehicles with ULA addresses, it is then possible to separate in-vehicle embedded networks into different subnets depending on real-time requirements, device types, services and more.

However, each isolated network has the possibility to be connected in the future. Administrators need to consider the following before deciding whether to use ULAs:

- If the network eventually connects to another isolated or private network, the potential for address collision arises. However, if the ULAs were generated in the standard way, this will not be a big problem.

- If the network eventually connects to the global Internet, then the operator will need to add a new global prefix and ensure that the address selection policy is properly set up on all interfaces.

If these further considerations are unacceptable for some reason, then the administrator needs to be careful about using ULAs in currently isolated networks.

Operational considerations:

- Prefix generation: Randomly generated according to the algorithms defined in [RFC4193] or manually assigned. Normally, automatic generation of the prefixes is recommended, following [RFC4193]. If there are some specific reasons that call for manual assignment, administrators have to plan the prefixes carefully to avoid collision.

- Prefix announcement: In some cases, networks may need to announce prefixes to each other. For example, in vehicle networks with infrastructure-less settings such as Vehicle-to-Vehicle (V2V) communication, prior knowledge of the respective prefixes is unlikely. Hence, a prefix announcement mechanism is needed to enable inter-vehicle communications based on IP. As one possibility, such announcements could rely on extensions to the Router Advertisement message of the Neighbor Discovery Protocol (e.g., [I-D.petrescu-autoconf-ra-based-routing] and [I-D.jhlee-mext-mnpp]).

4.2. Connected Networks

4.2.1. ULA-Only Deployment

In some situations, hosts and interior interfaces are assigned ULAs and not GUAs, but the network needs to communicate with the outside. Two models can be considered:

- Using Network Prefix Translation

  Network Prefix Translation (NPTv6) [RFC6296] is an experimental specification that provides a stateless one-to-one mapping between internal addresses and external addresses. The specification considers translating ULA prefixes into GUA prefixes as an use case. Although NPTv6 works differently from traditional stateful NAT/NAPT (which is discouraged in [RFC5902]), it introduces similar additional complexity to applications, which may cause applications to break.

Thus this document does not recommend the use of ULA+NPTv6. Rather, this document considers ULA+PA (Provider Aggregated) as a better approach to connect to the global network when ULAs are expected to be retained. The use of ULA+PA is discussed in detail in Section 4.2.2 below.

- Using Application-Layer Proxies

  The proxies terminate the network-layer connectivity of the hosts and associate separate internal and external connections.

  In some environments (e.g., information security sensitive enterprise or government), central control is exercised by allowing the endpoints to connect to the Internet only through a proxy. With IPv4, using private address space with proxies is an effective and common practice for this purpose, and it is natural to pick ULA as its counterpart in IPv6.

Benefits of using ULAs in this scenario:

- Allowing minimal management burden on address assignment for some specific environments.

Drawbacks:

- The serious disadvantages and impact on applications imposed by NATs have been well documented in [RFC2993] and [RFC3027]. Although NPTv6 is a mechanism that has fewer architectural problems than a traditional stateful Network Address Translator in an IPv6 environment [RFC6296], it still breaks end-to-end transparency and hence in general is not recommended by the IETF.

Operational considerations:

- Firewall deployment: [RFC6296] points out that an NPTv6 translator does not have the same security properties as a traditional NAT44, and hence needs be supplemented with a firewall if security at the boundary is an issue. The operator has to decide where to locate the firewall.

  - If the firewall is located outside the NPTv6 translator, then filtering is based on the translated GUA prefixes, and when the internal ULA prefixes are renumbered, the filtering rules do not need to be changed. However, when the GUA prefixes of the

NPTv6 are renumbered, the filtering rules need to be updated accordingly.).

- If the firewall is located inside the NPTv6 translator, the filtering is then based on the ULA prefixes, and the rules need to be updated correspondingly. There is no need to update when the NPTv6 GUA prefixes are renumbered.

4.2.2. ULAs along with PA Addresses

Two classes of network might need to use ULA with PA (Provider Aggregated) addresses:

- Home network. Home networks are normally assigned with one or more globally routed PA prefixes to connect to the uplink of an ISP. In addition, they may need internal routed networking even when the ISP link is down. Then ULA is a proper tool to fit the requirement. [RFC7084] requires the CPE to support ULA. Note: ULAs provide more benefit for multiple-segment home networks; for home networks containing only one segment, link-local addresses are better alternatives.

- Enterprise network. An enterprise network is usually a managed network with one or more PA prefixes or with a PI prefix, all of which are globally routed. The ULA can be used to improve internal connectivity and make it more resilient, or to isolate certain functions like OAM for servers.

Benefits of Using ULAs in this scenario:

- Separated local communication plane: for either home networks or enterprise networks, the main purpose of using ULAs along with PA addresses is to provide a logically local routing plane separated from the global routing plane. The benefit is to ensure stable and specific local communication regardless of the ISP uplink failure. This benefit is especially meaningful for the home network or for private OAM function in an enterprise.

- Renumbering: in some special cases such as renumbering, enterprise administrators may want to avoid the need to renumber their internal-only, private nodes when they have to renumber the PA addresses of the rest of the network because they are changing ISPs, because the ISP has restructured its address allocations, or for some other reason. In these situations, ULA is an effective tool for addressing internal-only nodes. Even public nodes can benefit from ULA for renumbering, on their internal interfaces. When renumbering, as [RFC4192] suggests, old prefixes continue to be valid until the new prefix(es) is(are) stable. In the process of adding new prefix(es) and deprecating old prefix(es), it is not easy to keep local communication disentangled from global routing plane change. If we use ULAs for local communication, the separated local routing plane can isolate the effects of global routing change.

Drawbacks:

- o Operational Complexity: there are some arguments that in practice the use of ULA+PA creates additional operational complexity. This is not a ULA-specific problem; the multiple-addresses-per-interface is an important feature of IPv6 protocol. Nevertheless, running multiple prefixes needs more operational consideration than running a single one.

Operational considerations:

- o Default Routing: connectivity may be broken if ULAs are used as default route. When using RIO (Route Information Option) in [RFC4191], specific routes can be added without a default route, thus avoiding bad user experience due to timeouts on ICMPv6 redirects. This behavior was well documented in [RFC7084] as rule ULA-5 "An IPv6 CE router MUST NOT advertise itself as a default router with a Router Lifetime greater than zero whenever all of its configured and delegated prefixes are ULA prefixes." and along with rule L-3 "An IPv6 CE router MUST advertise itself as a router for the delegated prefix(es) (and ULA prefix if configured to provide ULA addressing) using the "Route Information Option" specified in Section 2.3 of [RFC4191]. This advertisement is independent of having or not having IPv6 connectivity on the WAN interface.". However, it needs to be noticed that current OSes don’t all support [RFC4191].

- o SLAAC/DHCPv6 co-existing: Since SLAAC and DHCPv6 might be enabled in one network simultaneously; the administrators need to carefully plan how to assign ULA and PA prefixes in accordance with the two mechanisms. The administrators need to know the current issue of the SLAAC/DHCPv6 interaction (please refer to [I-D.ietf-v6ops-dhcpv6-slaac-problem] for details).

- o Address selection: As mentioned in [RFC5220], there is a possibility that the longest matching rule will not be able to choose the correct address between ULAs and global unicast addresses for correct intra-site and extra-site communication. [RFC6724] claims that a site-specific policy entry can be used to cause ULAs within a site to be preferred over global addresses.

- DNS relevant: if administrators choose not to do reverse DNS delegation inside of their local control of ULA prefixes, a significant amount of information about the ULA population may leak to the outside world. Because reverse queries will be made and naturally routed to the global reverse tree, so external parties will be exposed to the existence of a population of ULA addresses. [ULA-IN-WILD] provides more detailed situations on this issue. Administrators may need a split DNS to separate the queries from internal and external for ULA entries and GUA entries.

4.3. IPv4 Co-existence Considerations

Generally, this document does not consider IPv4 to be in scope. But regarding ULA, there is a special case needs to be recognized, which is described in Section 3.2.2 of [RFC5220]. When an enterprise has IPv4 Internet connectivity but does not yet have IPv6 Internet connectivity, and the enterprise wants to provide site-local IPv6 connectivity, a ULA is the best choice for site-local IPv6 connectivity. Each employee host will have both an IPv4 global or private address and a ULA. Here, when this host tries to connect to an outside node that has registered both A and AAAA records in the DNS, the host will choose AAAA as the destination address and the ULA for the source address according to the IPv6 preference of the default policy table defined in the old address selection standard [RFC3484]. This will clearly result in a connection failure. The new address selection standard [RFC6724] has corrected this behavior by preferring IPv4 than ULAs in the default policy table. However, there are still lots of hosts using the old standard [RFC3484], thus this could be an issue in real networks.

Happy Eyeballs [RFC6555] solves this connection failure problem, but unwanted timeouts will obviously lower the user experience. One possible approach to eliminating the timeouts is to deprecate the IPv6 default route and simply configure a scoped route on hosts (in the context of this document, only configure the ULA prefix routes). Another alternative is to configure IPv4 preference on the hosts, and not include DNS A records but only AAAA records for the internal nodes in the internal DNS server. Then outside nodes have both A and AAAA records and can be connected through IPv4 as default and internal nodes can always connect through IPv6. But since IPv6 preference is default, changing the default in all nodes is not suitable at scale.

5. General Considerations For Using ULAs

5.1. Do Not Treat ULA Equal to RFC1918

ULA and [RFC1918] are similar in some aspects. The most obvious one is as described in Section 3.1.3 that ULA provides an internal address independence capability in IPv6 that is similar to how [RFC1918] is commonly used. ULA allows administrators to configure the internal network of each platform the same way it is configured in IPv4. Many organizations have security policies and architectures based around the local-only routing of [RFC1918] addresses and those policies may directly map to ULA [RFC4864].

But this does not mean that ULA is equal to an IPv6 version of [RFC1918] deployment. [RFC1918] usually combines with NAT/NAPT for global connectivity. But it is not necessary to combine ULAs with any kind of NAT. Operators can use ULA for local communications along with global addresses for global communications (see Section 4.2.2). This is a big advantage brought by default support of multiple-addresses-per-interface feature in IPv6. (People may still have a requirement for NAT with ULA, this is discussed in Section 4.2.1. But people also need to keep in mind that ULA is not intentionally designed for this kind of use case.)

Another important difference is the ability to merge two ULA networks without renumbering (because of the uniqueness), which is a big advantage over [RFC1918].

5.2. Using ULAs in a Limited Scope

A ULA is by definition a prefix that is never advertised outside a given domain, and is used within that domain by agreement of those networked by the domain.

So when using ULAs in a network, the administrators need to clearly set the scope of the ULAs and configure ACLs on relevant border routers to block them out of the scope. And if internal DNS is enabled, the administrators might also need to use internal-only DNS names for ULAs and might need to split the DNS so that the internal DNS server includes records that are not presented in the external DNS server.

6. ULA Usages Considered Helpful

6.1. Used in Isolated Networks

As analyzed in Section 4.1, ULA is very suitable for isolated networks. Especially when there are subnets in the isolated network, ULA is a reasonable choice.

6.2. ULA along with PA

As described in Section 4.2.2, using ULAs along with PA addresses to provide a logically separated local plane can benefit OAM functions and renumbering.

6.3. Some Specific Use Cases

Along with the general scenarios, this section provides some specific use cases that could benefit from using ULA.

6.3.1. Special Routing

For various reasons the administrators may want to have private routing be controlled and separated from other routing. For example, in the business-to-business case described in [I-D.baker-v6ops-b2b-private-routing], two companies might want to use direct connectivity that only connects stated machines, such as a silicon foundry with client engineers that use it. A ULA provides a simple way to assign prefixes that would be used in accordance with an agreement between the parties.

6.3.2. Used as NAT64 Prefix

The NAT64 PREF64 is just a group of local fake addresses for the DNS64 to point traffic to a NAT64. Using a ULA prefix as the PREF64 easily ensures that only local systems can use the translation resources of the NAT64 system since the ULA is not intended to be globally routable. The ULA helps clearly identify traffic that is locally contained and destined to a NAT64. Using ULA for PREF64 is deployed and it is an operational model.

But there is an issue needs to be noted. The NAT64 standard [RFC6146] specifies that the PREF64 should align with [RFC6052], in which the IPv4-Embedded IPv6 Address format was specified. If we pick a /48 for NAT64, it happens to be a standard 48/ part of ULA (7bit ULA well-known prefix+ 1 "L" bit + 40bit Global ID). Then the 40bit of ULA is not violated by being filled with part of the 32bit IPv4 address. This is important, because the 40bit assures the uniqueness of ULA. If the prefix is shorter than /48, the 40bit would be violated, and this could cause conformance issues. But it is considered that the most common use case will be a /96 PREF64, or even /64 will be used. So it seems this issue is not common in current practice.

It is most common that ULA PREF64 will be deployed on a single internal network, where the clients and the NAT64 share a common internal network. ULA will not be effective as PREF64 when the access network must use an Internet transit to receive the translation service of a NAT64 since the ULA will not route across the Internet.

According to the default address selection table specified in [RFC6724], the host would always prefer IPv4 over ULA. This could be a problem in NAT64-CGN scenario as analyzed in Section 8 of [RFC7269]. So administrators need to add additional site-specific address selection rules to the default table to steer traffic flows going through NAT64-CGN. However, updating the default policy tables in all hosts involves significant management cost. This may be possible in an enterprise (using a group policy object, or other configuration mechanisms), but it is not suitable at scale for home networks.

6.3.3. Used as Identifier

ULAs could be self-generated and easily grabbed from the standard IPv6 stack. And ULAs don’t need to be changed as the GUA prefixes do. So they are very suitable to be used as identifiers by the up layer applications. And since ULA is not intended to be globally routed, it is not harmful to the routing system.

Such kind of benefit has been utilized in real implementations. For example, in [RFC6281], the protocol BTMM (Back To My Mac) needs to assign a topology-independent identifier to each client host according to the following considerations:

- TCP connections between two end hosts wish to survive in network changes.

- Sometimes one needs a constant identifier to be associated with a key so that the Security Association can survive the location changes.

It needs to be noticed again that in theory ULA has the possibility of collision. However, the probability is desirably small enough and can be ignored in most cases when ULAs are used as identifiers.

7. Security Considerations

Security considerations regarding ULAs, in general, please refer to the ULA specification [RFC4193]. Also refer to [RFC4864], which shows how ULAs help with local network protection.

As mentioned in Section 4.2.2, when using NPTv6, the administrators need to know where the firewall is located to set proper filtering rules.

Also as mentioned in Section 4.2.2, if administrators choose not to do reverse DNS delegation inside their local control of ULA prefixes, a significant amount of information about the ULA population may leak to the outside world.

8. IANA Considerations

This memo has no actions for IANA.

9. Acknowledgements

Many valuable comments were received in the IETF v6ops WG mail list, especially from Cameron Byrne, Fred Baker, Brian Carpenter, Lee Howard, Victor Kuarsingh, Alexandru Petrescu, Mikael Abrahamsson, Tim Chown, Jen Linkova, Christopher Palmer Jong-Hyouk Lee, Mark Andrews, Lorenzo Colitti, Ted Lemon, Joel Jaeggli, David Farmer, Doug Barton, Owen Delong, Gert Doering, Bill Jouris, Bill Cerveny, Dave Thaler, Nick Hilliard, Jan Zorz, Randy Bush, Anders Brandt, , Sofiane Imadali and Wesley George.

Some test of using ULA in the lab was done by our research partner BNRC-BUPT (Broad Network Research Centre in Beijing University of Posts and Telecommunications). Thanks for the work of Prof. Xiangyang Gong and student Dengjia Xu.

Tom Taylor did a language review and revision throught the whole document. The authors appreciate a lot for his help.

This document was produced using the xml2rfc tool [RFC2629] (initially prepared using 2-Word-v2.0.template.dot.).

10. References

10.1. Normative References

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

[RFC2629] Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629, June 1999.

[RFC4193] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast Addresses", RFC 4193, October 2005.

10.2. Informative References

[I-D.baker-v6ops-b2b-private-routing]
Baker, F., "Business to Business Private Routing", draft-baker-v6ops-b2b-private-routing-00 (work in progress), July 2007.

[I-D.ietf-v6ops-dhcpv6-slaac-problem]
Liu, B., Jiang, S., Bonica, R., Gong, X., and W. Wang, "DHCPv6/SLAAC Address Configuration Interaction Problem Statement", draft-ietf-v6ops-dhcpv6-slaac-problem-03 (work in progress), October 2014.

[I-D.jhlee-mext-mnpp]
Tsukada, M., Ernst, T., and J. Lee, "Mobile Network Prefix Provisioning", draft-jhlee-mext-mnpp-00 (work in progress), October 2009.

[I-D.petrescu-autoconf-ra-based-routing]
Petrescu, A., Janneteau, C., Demailly, N., and S. Imadali, "Router Advertisements for Routing between Moving Networks", draft-petrescu-autoconf-ra-based-routing-05 (work in progress), July 2014.

[MIL-STD-1397]
"Military Standard, Input/Output Interfaces, Standard Digital Data, Navy Systems (MIL-STD-1397B), 3 March 1989".

[RFC1918] Rekhter, Y., Moskowitz, R., Karrenberg, D., Groot, G., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, February 1996.

[RFC2993] Hain, T., "Architectural Implications of NAT", RFC 2993, November 2000.

[RFC3027] Holdrege, M. and P. Srisuresh, "Protocol Complications with the IP Network Address Translator", RFC 3027, January 2001.

[RFC3484] Draves, R., "Default Address Selection for Internet Protocol version 6 (IPv6)", RFC 3484, February 2003.

[RFC3879] Huitema, C. and B. Carpenter, "Deprecating Site Local Addresses", RFC 3879, September 2004.

[RFC4191] Draves, R. and D. Thaler, "Default Router Preferences and More-Specific Routes", RFC 4191, November 2005.

[RFC4192] Baker, F., Lear, E., and R. Droms, "Procedures for Renumbering an IPv6 Network without a Flag Day", RFC 4192, September 2005.

[RFC4864] Van de Velde, G., Hain, T., Droms, R., Carpenter, B., and E. Klein, "Local Network Protection for IPv6", RFC 4864, May 2007.

[RFC5220] Matsumoto, A., Fujisaki, T., Hiromi, R., and K. Kanayama, "Problem Statement for Default Address Selection in Multi-Prefix Environments: Operational Issues of RFC 3484 Default Rules", RFC 5220, July 2008.

[RFC5902] Thaler, D., Zhang, L., and G. Lebovitz, "IAB Thoughts on IPv6 Network Address Translation", RFC 5902, July 2010.

[RFC6052] Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", RFC 6052, October 2010.

[RFC6146] Bagnulo, M., Matthews, P., and I. van Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", RFC 6146, April 2011.

[RFC6281] Cheshire, S., Zhu, Z., Wakikawa, R., and L. Zhang, "Understanding Apple’s Back to My Mac (BTMM) Service", RFC 6281, June 2011.

[RFC6296] Wasserman, M. and F. Baker, "IPv6-to-IPv6 Network Prefix Translation", RFC 6296, June 2011.

[RFC6555] Wing, D. and A. Yourtchenko, "Happy Eyeballs: Success with Dual-Stack Hosts", RFC 6555, April 2012.

[RFC6724] Thaler, D., Draves, R., Matsumoto, A., and T. Chown, "Default Address Selection for Internet Protocol Version 6 (IPv6)", RFC 6724, September 2012.

[RFC7084] Singh, H., Beebee, W., Donley, C., and B. Stark, "Basic Requirements for IPv6 Customer Edge Routers", RFC 7084, November 2013.

[RFC7269] Chen, G., Cao, Z., Xie, C., and D. Binet, "NAT64 Deployment Options and Experience", RFC 7269, June 2014.

[RS-485] "Electronic Industries Association (1983). Electrical Characteristics of Generators and Receivers for Use in Balanced Multipoint Systems. EIA Standard RS-485.".

[SCADA] "Boyer, Stuart A. (2010). SCADA Supervisory Control and Data Acquisition. USA: ISA - International Society of Automation.".

[ULA-IN-WILD]
"G. Michaelson, "conference.apnic.net/data/36/apnic-36-ula_1377495768.pdf"".

Authors’ Addresses

Bing Liu
Huawei Technologies
Q14, Huawei Campus, No.156 Beiqing Road
Hai-Dian District, Beijing, 100095
P.R. China

Email: leo.liubing@huawei.com

Sheng Jiang
Huawei Technologies
Q14, Huawei Campus, No.156 Beiqing Road
Hai-Dian District, Beijing, 100095
P.R. China

Email: jiangsheng@huawei.com

V6OPS B. Liu
Internet-Draft Huawei Technologies
Intended status: Informational R. Bonica
Expires: April 30, 2015 Juniper Networks
T. Yang
China Mobile
October 27, 2014

DHCPv6/SLAAC Interaction Operational Guidance
draft-liu-v6ops-dhcpv6-slaac-guidance-03

Abstract

The IPv6 Neighbor Discovery (ND) Protocol [RFC4861] specifies an ICMPv6 Router Advertisement (RA) message. The RA message contains three flags that indicate which address autoconfiguration mechanisms are available to on-link hosts. These are the M, O and A flags. The M, O and A flags are all advisory, not prescriptive.

In [I-D.ietf-v6ops-dhcpv6-slaac-problem], test results show that in several cases the M, O and A flags elicit divergent host behaviors, which might cause some operational problems. This document aims to provide some operational guidance to eliminate the impact caused by divergent host behaviors as much as possible.

Status of This Memo

This Internet-Draft will expire on April 30, 2015.

Copyright Notice

Table of Contents

1. Introduction

The IPv6 Neighbor Discovery (ND) Protocol [RFC4861] specifies an ICMPv6 Router Advertisement (RA) message. The RA message contains three flags that indicate which address autoconfiguration mechanisms are available to on-link hosts. These are the M, O and A flags. The M, O and A flags are all advisory, not prescriptive.

In [I-D.ietf-v6ops-dhcpv6-slaac-problem], test results show that in several cases the M, O and A flags elicit divergent host behaviors, which might cause some operational problems. This document aims to provide some operational guidance to eliminate the impact caused by divergent host behaviors as much as possible.

This document does not intent to cover the topic of selection between RA and DHCPv6 [RFC3315] for the overlapped functions. There always are arguments about what should be done through RA options or through DHCPv6 options. For this general issue, draft [I-D.yourtchenko-ra-dhcpv6-comparison] could be referred.

2. Operational Guidance

2.1. Always Turn RAs On

Currently, turning RAs on is actually a basic requirement for running IPv6 networks since only RAs could advertise default route(s) for the end nodes. And if the nodes want to communicate with each other on the same link via DHCPv6-configured addresses, they also need to be advertised with L flag set in RAs. So for current networks, an IPv6 network could NOT run without RAs, unless the network only demands a communication via link-local addresses.

2.2. Guidance for DHCPv6/SLAAC Provisioning Scenarios

2.2.1. DHCPv6-only

In IPv4, there is only one method (DHCPv4) for automatically configuring the hosts. Many network operations/mechanisms, especially in enterprise networks, are built around this central-managed model. So it is reasonable for people who are accustomed to DHCPv4-only deployment still prefer DHCPv6-only in IPv6 networks. Besides, some networks just prefer central management of all IP addressing. These networks may want to assign addresses only via DHCPv6.

This can be accomplished by sending RAs that indicate DHCPv6 is available (M=1), installing DHCPv6 servers or DHCPv6 relays on all links, and setting A=0 in the Prefix Information Options of all prefixes in the RAs. (Instead of forcing the A flag off, simply not including any PIO in RAs could also make the same effect). But before doing this, the administrators need to be sure that every node in their intended management scope supports DHCPv6.

Note that RAs are still necessary in order for hosts to be able to use these addresses. This is for two reasons:

- If there is no RA, some hosts will not attempt to obtain address configuration via DHCPv6 at all.

- DHCPv6 can assign addresses but not routing. Routing can be implemented on hosts by means of accepting and implementing information from RA messages containing default-route, Prefix Information Option with O=1, or Route Information Option, or by configuring manual routing. Without routing, IPv6 addresses won’t be used for communication outside the host. Thus, for example, if there is no RA and no static routing, then addresses assigned by DHCPv6 cannot be used even for communication between hosts on the same link.

Also note that unlike SLAAC [RFC4862], DHCPv6 is not a strict requirement for IPv6 hosts [RFC6434], and some nodes do not support DHCPv6. Thus, this model can only be used if all the hosts that need IPv6 connectivity support DHCPv6.

2.2.2. SLAAC-only

In contrast with DHCPv6-only, some scenarios might be suitable for SLAAC-only which allows minimal administration burden and node capability requirement.

The administrators MUST turn the A flag on, and MUST turn M flag off. Note that some platforms (e.g. Windows 8) might still initiate DHCPv6 session regardless of M flag off. But since there is no DHCPv6 service available, the only problem is that there would be some unnecessary traffic.

2.2.3. DHCPv6/SLAAC Co-existence

- Scenarios of DHCPv6/SLAAC Co-existence
    * For provisioning redundancy: If the administrators want all nodes at least could configure a global scope address, then they could turn A flag and M flag both on in case some nodes only support one of the mechanisms. For example, some hosts might only support SLAAC; while some hosts might only support DHCPv6 due to manual/mistaken configurations.
    * For different provisioning: the two address configuration mechanisms might provide two addresses for the nodes respectively. For example, SLAAC-configured address is for basic connectivity and another address configured by DHCPv6 is for a specific service.
- Cautions
    * Notice that enabling both DHCPv6 and SLAAC would cause one host to configure more IPv6 addresses. Typically, there would be one more DHCPv6-configured address than SLAAC-only configuration; and two more addresses based on SLAAC and privacy extension than DHCPv6-only configuration. Too many addresses might cause ND cache overflow problem in some

situations (please refer to Section 3.4 of [I-D.liu-v6ops-running-multiple-prefixes] for details).

* For provisioning redundancy scenario, there is a concern that SLAAC/DHCPv6 addresses based on the same prefix might cause some applications confusing. [Open Question] Call for real experiences on this issues.

* Besides address configuration, DNS can also be configured both by SLAAC and DHCPv6. If the DNS information in RAs and DHCPv6 are different, the host might confuse. So in terms of operation, the operators should make sure DNS configuration in RAs and DHCPv6 are the same.

2.3. Guidance for Renumbering

This document only considers the renumbering cases where DHCPv6/SLAAC interaction is involved. These renumbering operations need the A/M flags transition which might cause unpredictable host behaviors. Two renumbering cases are discussed as the following.

2.3.1. Adding a New Address from another Address Configuration Mechanisms

o Adding a DHCPv6 Address for a SLAAC-configured Host

As discussed in Section 2.2.3, some operating systems that having configured SLAAC addersses would NOT care about the newly added DHCPv6 provision unless the current SLAAC address lifetime is expired. In theory, one possible way is to stop advertising RAs and wait the SLAAC addresses expired (this makes the hosts return to the initial stage), then advertise RAs again with the M flag set, so that the host would configure SLAAC and DHCPv6 addresses simultaneously. However, there would be some outage period during this operation, which might be unacceptable for many situations. Thus, It is better for the administrators to carefully plan the network provisioning so that to make SLAAC and DHCPv6 available simultaneously (through RA with M=1) at the initial stage rather than configuring one and then configuring another.

o Adding a SLAAC Address for a DHCPv6-configured Host

As tested in [I-D.ietf-v6ops-dhcpv6-slaac-problem].), current mainstream operating systems all support this renumbering operation. The only thing need to care about is to make sure the M flag is on in the RAs, since some operating systems would immediately release the DHCPv6 addresses if M flag is off.

2.3.2. Switching one Address Configuration Mechanism to another

- DHCPv6 to SLAAC

  This operation is supported by all the tested operating systems in [I-D.ietf-v6ops-dhcpv6-slaac-problem]. However, the behaviors are different. As said above, if A flag is on while M flag is off, a flash switching renumbering would happen on some operating systems. So while turning the A flag on, it is recommended to retain the M flag on and stop the DHCPv6 server to response the renew messages so that the DHCPv6 addresses could be released when the lifetimes expired.

- SLAAC to DHCPv6

  This operation is also supported by all the tested operating systems. And the behaviors are the same since no operating systems would immediatly release the SLAAC addresses when A flag is off. However, for safe operation, while turning the M flag on, it is also recommended to retain the A flag on and stop advertising RAs so that the SLAAC addresses could be released when the lifetimes expired.

3. Security Considerations

No more security considerations than the Neighbor Discovery protocol [RFC4861].

4. IANA Considerations

This draft does not request any IANA action.

5. Acknowledgements

Valuable comments were received from Sheng Jiang and Brian E Carpenter to initiate the draft. Some texts in Section 2.2.1 were based on Lorenzo Colitti and Mikael Abrahamsson’s proposal. There were also comments from Erik Nordmark, Ralph Droms, John Brzozowski, Andrew Yourtchenko and Wesley George to improve the draft. The authors would like to thank all the above contributors.

This document was produced using the xml2rfc tool [RFC2629]. (This document was initiallly prepared using 2-Word-v2.0.template.dot.)

6. References

6.1. Normative References

[RFC2629] Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629, June 1999.

[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.

[RFC6434] Jankiewicz, E., Loughney, J., and T. Narten, "IPv6 Node Requirements", RFC 6434, December 2011.

6.2. Informative References

[I-D.ietf-v6ops-dhcpv6-slaac-problem] Liu, B., Jiang, S., Bonica, R., Gong, X., and W. Wang, "DHCPv6/SLAAC Address Configuration Interaction Problem Statement", draft-ietf-v6ops-dhcpv6-slaac-problem-02 (work in progress), October 2014.

[I-D.liu-v6ops-running-multiple-prefixes] Liu, B., Jiang, S., and Y. Bo, "Considerations for Running Multiple IPv6 Prefixes", draft-liu-v6ops-running-multiple-prefixes-02 (work in progress), October 2014.

[I-D.yourtchenko-ra-dhcpv6-comparison] Yourtchenko, A., "A comparison between the DHCPv6 and RA based host configuration", draft-yourtchenko-ra-dhcpv6-comparison-00 (work in progress), November 2013.

[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.

Authors’ Addresses

Bing Liu
Huawei Technologies
Q14, Huawei Campus, No.156 Beiqing Road
Hai-Dian District, Beijing, 100095
P.R. China

Email: leo.liubing@huawei.com

Ron Bonica
Juniper Networks
Sterling, Virginia
20164
USA

Email: rbonica@juniper.net

Tianle Yang
China Mobile
32, Xuanwumenxi Ave.
Xicheng District, Beijing 100053
P.R. China

Email: yangtianle@chinamobile.com

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: June 6, 2014

J. Jaeggli
Zynga
L. Colitti
W. Kumari
Google
E. Vyncke
Cisco
M. Kaeo
Double Shot Security
T. Taylor, Ed.
Huawei Technologies

December 3, 2013

Why Operators Filter Fragments and What It Implies
draft-taylor-v6ops-fragdrop-02

Abstract

This memo was written to make application developers and network operators aware of the significant possibility that IPv6 packets containing fragmentation extension headers may fail to reach their destination. Some protocol or application assumptions about the ability to use messages larger than a single packet may accordingly not be supportable in all networks or circumstances.

This memo provides observational evidence for the dropping of IPv6 fragments along a significant number of paths, explores the operational impact of fragmentation and the reasons and scenarios where drops occur, and considers the effect of fragment drops on applications where fragmentation is known to occur, particularly including DNS.

Status of This Memo

This Internet-Draft will expire on June 6, 2014.

Copyright Notice

Table of Contents

1. Introduction

Measurements of whether Internet Service Providers and edge networks deliver IPv6 fragments to their destination reveal that for IPv6 in particular, fragments are being dropped along a substantial number of paths. The filtering of IPv6 datagrams with fragmentation headers is presumed to be a non-issue in the core of the Internet, where fragments are routed just like any other IPv6 datagram. However, fragmentation can creates operational issues at the edges of the Internet that may lead to administratively imposed filtering or inadvertent failure to deliver the fragment to the end-system or application.

Section 2 begins with some observations on how often IPv6 fragment loss occurs in practice. We go on to look at the operational reasons for filtering fragments, a key aspect of which is the limitations they expose in the application of security policy, at resource bottlenecks and in forwarding decisions. Section 2.2 then looks at the impact on key applications, particularly DNS.

In the longer run, as network operators gain a better understanding of the risks and non-risks of fragmentation and as middlebox, customer premise equipment (CPE), and host implementations improve, we believe that some incidence of fragment dropping currently required will diminish. Some of the justifications for filtering will persist in the long-term, and application developers and network operators must remain aware of the implications.

This document deliberately refrains from discussing possible responses to the problem posed by the dropping of IPv6 fragments. Such a discussion will quickly turn up a number of possibilities, application-specific or more general; but the amount of time needed to specify and deploy a given resolution will be a major constraint in choosing amongst them. In any event, that discussion is likely to proceed in multiple directions, occur in different areas and is therefore considered beyond the scope of this memo.

2. Observations and Rationale

[Blackhole] is a good public reference for some empirical data on IPv6 fragment filtering. It describes experiments run to determine the incidence and location of ICMP Packet Too Big and fragment filtering. The authors used fragmented DNS packets to determine the latter, setting the servers to an IPv6 minimum of 1280 bytes to avoid any PMTU issues. The tests found for IPv6 that filtering appeared to be occurring on some 10% of the tested paths. The filtering appeared to be located at the edge (enterprise and customer networks) rather than in the core.

2.1. Possible Causes

Why does such filtering happen? One cause is non-conforming implementations in CPE and low-end routers. Some network managers filter fragments on principle, thinking this is an easier way to deter realizable attacks utilizing IPv6 fragments without thinking of other network impacts, similar to the practice of filtering ICMP Packet Too Big. Both implementations and management should improve over time, reducing the problem somewhat.

Some filtering and dropping of fragments is known to be done for hardware, performance, or topological considerations.

2.1.1. Stateful inspection

Stateful inspection devices or destination hosts can readily experience resource exhaustion if they are flooded with fragments that are not followed in a timely manner by the remaining fragments of the original datagram. Holding fragments for reassembly even on end-system firewalls can readily result in an effective denial of service by memory and CPU exhaustion even if techniques, such as virtual re-assambly exist.

2.1.2. Stateless ACLs

Stateless ACLs at layer 4 and up may be difficult to apply to fragments other than the first one in which enough of the upper layer header is present. As [Attacks] demonstrates, inconsistencies in reassembly logic between middleboxes or CPEs and hosts can cause fragments to be wrongfully discarded, or can allow exploits to pass undetected through middleboxes. Stateless load balancing schemes may hash fragmented datagrams from the same flow to different paths because the 5-tuple may be available on only the initial fragment. While rehashing has the possibility of reordering packets in ISP cores it is not disastrous. However, in front of a stateful inspection device, load balancer tier, or anycast service instance, where headers other than the L3 header -- for example, the L4 header, interface index (for traffic already rehashed onto different paths), DS fields -- are considered as part of the hash, rehashing may result in the fragments being delivered to different end-systems

2.1.3. Performance considerations

Leaving aside these incentives towards fragment dropping, other considerations may weigh on the operator’s mind. One example cited on the NANOG list was that of a router where fragment processing was done by the control plane processor rather than in the forwarding plane hardware, with a consequent hit on performance.

2.1.4. Other considerations

Another incentive toward dropping of fragments is the disproportionate number of software errors still being encountered in fragment processing. Since this code is exercised less frequently than the rest of the stack, bugs remain longer in the code before they are detected. Some of these software errors can introduce vulnerabilities subject to exploitation. It is common practice [RFC6192] to recommend that control-plane ACLs protecting routers and network devices be configured to drop all fragments.

2.1.5. Conclusions

Operators weigh the risks associated with each of the considerations just enumerated, and come up with the most suitable policy for their circumstances. It is likely that at least some operators will find it desirable to drop fragments in at least some cases.

The IETF and operators can help this effort by identifying specific classes of fragments that do not represent legitimate use cases and hence should always be dropped. Examples of this work are given by [RFC6946] and [I-D.ietf-6man-oversized-header-chain]. The problem of inconsistent implementations may also be mitigated by providing further advice on the more difficult points. However, some cases will remain where legitimate fragments are discarded for legitimate reasons. The potential problems these cases pose for applications is our next topic.

2.2. Impact on Applications

Some applications can live without fragmentation, some cannot. UDP DNS is one application that has the potential to be impacted when fragment dropping occurs. EDNS0 extensions [RFC2671] allow for responses in UDP PDUs that are greater than 512 bytes. Particularly with DNSSEC [RFC4033], responses may be larger than the link MTU and fragmentation would therefore occur at the sending host in order to respond using UDP. The current choices open to the operators of DNS servers in this situation are to defer deployment of DNSSEC, fragment responses, or use TCP if there are cases where the rrset would be expected to exceed the MTU. The use of fallback to TCP will impose a major resource and performance hit and increases vulnerability to denial of service attacks.

Other applications, such as the Network File System, NFS, are also known to fragment large UDP packets for datagrams larger than the MTU. NFS is most often restricted to the internal networks of organizations. In general, managing NFS connectivity should not be impacted by decisions mananging fragment drops at network borders or end-systems.

3. Acknowledgements

The authors of this document would like to thank the RIPE Atlas project and NLNetlabs whose conclusions ignited this document.

4. IANA Considerations

This memo includes no request to IANA.

5. Security Considerations

The potential for denial of service attacks, as well as limitations inherent in upper-layer filtering when dealing with non-initial fragments are significant issues under consideration by operators and end-users filtering fragments. This document does not offer alternative solutions to that problem, it does describe the impact of those filtering practices.

6. Informative References

[Attacks] Atlasis, A., "Attacking IPv6 Implementation Using Fragmentation", March 2012.

http://media.blackhat.com/bh-eu-12/Atlasis/bh-eu-12-Atlasis-Attacking_IPv6-WP.pdf

[Blackhole] de Boer, M. and J. Bosma, "Discovering Path MTU black holes on the Internet using RIPE Atlas", July 2012.

http://www.nlnetlabs.nl/downloads/publications/pmtu-black-holes-msc-thesis.pdf

[I-D.ietf-6man-oversized-header-chain] Gont, F., Manral, V., and R. Bonica, "Implications of Oversized IPv6 Header Chains", draft-ietf-6man-oversized-header-chain-08 (work in progress), October 2013.

[RFC2671] Vixie, P., "Extension Mechanisms for DNS (EDNS0)", RFC 2671, August 1999.

[RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, March 2005.

[RFC6192] Dugal, D., Pignataro, C., and R. Dunn, "Protecting the Router Control Plane", RFC 6192, March 2011.

[RFC6946] Gont, F., "Processing of IPv6 "Atomic" Fragments", RFC 6946, May 2013.

Authors’ Addresses

Joel Jaeggli
Zynga
630 taylor ct #10
Mountain View, CA 94043
USA

Email: jjaeggli@zynga.com

Lorenzo Colitti
Google

Email: lorenzo@google.com

Warren Kumari
Google
1600 Amphitheatre Parkway
Mountain View, CA 94043
USA

Email: warren@kumari.net

Eric Vyncke
Cisco
De Kleetlaan 6A
Diegem 1831
Belgium

Email: evyncke@cisco.com

Merike Kaeo
Double Shot Security

Email: merike@doubleshotsecurity.com

Tom Taylor (editor)
Huawei Technologies
Ottawa, Ontario
Canada

Email: tom.taylor.stds@gmail.com

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: August 18, 2014

E. Vyncke, Ed.
P. Thubert
E. Levy-Abegnoli
A. Yourtchenko
Cisco

February 14, 2014

Why Network-Layer Multicast is Not Always Efficient At Datalink Layer

draft-vyncke-6man-mcast-not-efficient-01

Abstract

Several IETF protocols (IPv6 Neighbor Discovery for example) rely on IP multicast in the hope to be efficient with respect to available bandwidth and to avoid generating interrupts in the network nodes. On some datalink-layer network, for example IEEE 802.11 WiFi, this is not the case because of some limitations in the services offered by the datalink-layer network. This document lists and explains all the potential issues when using network-layer multicast over some datalink-layer networks.

Status of This Memo

This Internet-Draft will expire on August 18, 2014.

Copyright Notice

Table of Contents

1. Introduction

Several IETF protocols rely on the use of link-local scoped IP multicast in the hope of reducing traffic over the underlying datalink network and generating less operating systems interrupts for the receiving nodes. For example, IPv6 Neighbor Discovery [RFC4861] uses link-local multicast to:

- advertise the presence of a router by sending router advertisement to IPv6 address link-local multicast address (LLMA), ff02::1, whose members are only the IPv6 nodes but per [RFC4291] section 3 those messages must be forwarded on all ports. This IPv6 LLMA is mapped to the Ethernet Multicast Address (EMA) 33:33:00:00:00:01;

- solicit the data-link layer address of an adjacent on-link node by sending a neighbor solicitation to the solicited-node multicast address corresponding to the target address such as ff02:0:0:0:0:1:ffXX:XXXX (where the last 24 bits are the last 24 bits of the target address) as described in [RFC4291]. This IPv6 LLMA is mapped to the EMA 33:33:ff:XX:XX:XX.

2. Issue on Wired Ethernet Network

Most switch vendors implement MLD snooping [RFC4541] in order to forward multicast frames only to switch ports where there is a member of the IPv6 multicast group. This optimization works by installing hardware forwarding states in the switch. As there is a finite amount of memory in the switches, especially when the memory is used by the data plane forwarding, there is also a limit to the number of MLD optimization states i.e. a limit to the number of IPv6 multicast groups that can be optimized by the switch; frames destined to groups without such a state are flooded on all ports in the same datalink domain, and generally the use of MLD snooping is reserved to groups with a scope wider than link local.

With IPv6, all nodes have usually at least two IPv6 addresses: a link-local and a global address. If both addresses are based on EUI-64, then they share the same 24 least-significant bits, hence there is only one solicited-node multicast address per node. Else, there is a high probability that the 24 least-significant bits are different, hence requiring the membership to two solicited-node multicast addresses. If a switch uses MLD snooping to install hardware-optimized multicast forwarding states for LLMA, then the switch installs two hardware-optimized states per node as EUI-64 addresses are no more commonly used. If privacy extension addresses [RFC4941] are used, then every node can have multiple IPv6 global addresses, most of which are not based on EUI-64, a large switch fabric will have to support multiple times more states for multicast EMA than it does for unicast addresses, resulting in an excessive amount of resources in each individual switch to be built at an affordable price.

Therefore, due to cost reason, the multicast optimization by MLD snooping of solicited-node LLMA is disabled on most Ethernet switches. This means wasting:

- the switch bandwidth as it works as a full-duplex hub;
- the nodes CPU as all nodes will have to receive the multicast frame (if their network adapter is not optimized to support MAC multicast) and quickly drop it.

A special mention must be paid when a layer-2 domain includes legacy devices working on at 10 Mbps half-duplex; for example, in hospitals having old equipments dated back of 1990. For this case, it takes only 100 300-byte frames per second to already utilize the media to 2.4 % not to mention that the NIC and the processor have to process those frames and that the processor is probably also dated from 1990...

It is unclear what the impact is on virtual machines with different MAC addresses and different IPv6 address connected with a virtual layer-2 switch hosted on a single physical server... The MLD snooping done by the virtual switch will consume CPU by the hypervisor, hence, also reducing the amount of CPU available for the virtual machines.

Leveraging MLD snooping to save layer-2 switches from flooding link-local multicast messages carries additional challenges. Unsolicited MLD reports are usually sent once (when link comes up) and not acknowledged. There exist a retransmission mechanism, but it is not generally deployed, and it does not guarantee that subsequent retransmission won’t also get lost. The switch could easily end up with incomplete forwarding states for a given group, with some of the listeners ports, but not all (much worse than no state at all). As the switch does not know one of its forwarding entry is incomplete, it can’t fall back to broadcasting. As ordinary MLD routers, the switch could query reports on a periodic basis. However, it is not practical for layer-2 access switches to send periodic general MLD queries to maintain forwarding states accuracy for at least 2 reasons:

- The queries must be sourced with a link-local IPv6 address, one per link, and, for many practical reasons, layer-2 switches don’t have such address on each link (vlan) they operate on.

- Since address resolution uses a multicast group, and may happen quite frequently on the link, in order to avoid black holing resolution, the interval for a switch to issue MLD general query would have to be very small (a few seconds). These MLD queries are themselves sent to a multicast group that all nodes would need to get. That would completely defeat the purpose of reducing multicast traffic towards end nodes.

3. Issues on IEEE 802.11 Wireless Network

3.1. Multicast over Wireless

Wireless networks are a shared half-duplex media: when one station transmits, then all others must be silent. A multicast or broadcast transmission from an AP is physically transmitted to all WiFi cliens (STAs) and no other node can use the wireless medium at that time. This is the first issue with the use of wireless for multicast: the medium access behaves as a Ethernet hub.

Depending on distance and radio propagation, different wireless clients may use different transmission encodings and data rates. A lower data rate effectively locks the medium for a longer time per bit. In order to reach all nodes, and considering that multicast and broadcast frames are not protected by ARQ (retries), the AP is constrained to transmit all multicast or broadcast frames at the lowest rate possible, which in practice is often translated to rates as low as 1 Mbps or 6 Mbps, even when the unicast rate can reach a hundred of Mbps and above. It results that sending a single multicast frame can consume as much bandwidth as dozens of unicast frames. Table Table 1 provides some example values of the bandwidth used by multicast frames transmitted from the AP (i.e. not counting the original multicast frame transmitted by the WiFi client to the AP when he source is effectively wireless).

Lowest WiFi rate	Highest WiFi rate	Mcast frame %-age	WiFi Utilization by Mcast
1 Mbps	11 Mbps	1 %	9 %
6 Mbps	54 Mbps	1 %	9 %
6 Mbps	54 Mbps	5 %	45 %
6 Mbps	54 Mbps	10 %	90 %

Table 1: Multicast WiFi Usage

If multiple APs cover the same wireless LAN, then the multicast frames must be transmitted by all APs to all their WiFi clients.

Communication of a multicast frame by a WiFi client requires three steps:

1. The WiFi client sends a datalink unicast frame to the AP at its maximum possible rate.

2. The WiFi AP forwards this frame on its wired interface and broadcasts it (as explained above) to all its WiFi clients. If there are multiple APs on the same datalink domain, then, all APs also broadcast this multicast frame to their WiFi clients.

3. A WiFi NIC that implements the STA in the client filters the frames that are effectively expected by this device based on destination address.

Another side effect of multicast frames is that there cannot be an acknowledgement mechanism (ARQ) similar to that used for unicast frame, therefore frames can be missed and NDP does not take this non negligible packet loss into account. This could have a negative impact for Duplicate Address Detection (DAD) if the multicast NS or the multicast NA with override are lost. Assuming a error rate of 8%

of corrupted frame, this means a 8% chance of loosing a complete frame, this means a 16% chance of not detecting a duplicate address.

For a well-distributed multicast group where relatively few devices actually participate to any given group, there should be no transmission at all if none of the clients expects the multicast destination address, and there should be very few unicast but fast transmissions to the limited set of interest STAs when there is effectively a match in the set of associated devices. But there is no mechanism in place to ensure that functionality.

3.2. Host Sleep Mode

When a sleeping host wakes up by a user interaction, it cannot determine whether it has moved to another network (SSID are not unique), hence, it has to send a multicast Router Solicitation (which triggers a Router Advertisement message from all adjacent routers) and the mobile host has to do Duplicate Address Detection for its link-local and global addresses, thus means transmitting at least two multicast Neighbour Solicitation messages which will be repeated by the AP to all other WiFi clients.

This process creates a lot of multicast packets:

- one multicast Router Solicitation from the WiFi client, which is received by the AP and if the AP is not optimized, then the Router Solitication is broadcasted again over the wireless link;

- one multicast Neighbor Solitication for the host LLA from the WiFi client, which is received by the AP and if the AP is not optimized, the message is transmitted back over the wireless link;

- per global address (usually 1 or 2 depending on whether privacy extension is active), same behavior as above.

In conclusion and in the good case of not having privacy extension, this means 6 WiFi broadcast packets plus the unicast replies on each wake-up of the device. Assuming a packet size of 80 bytes, this translates into about 120 bytes to take into account the WiFi frame format which is larger than the usual Ethernet frame, the table Table 2 gives some result of the WiFi utilization just for the multicast part of the wake-up of sleeping devices... This does not take into account the rest of the multicast utilization used by RS, RA, NS, NA, MLD, ... and the associated unicast traffic.

WiFi Clients	Wake-up Cycle	Mcast packet/sec	Mcast bit/sec	Lowest WiFi Rate	Mcast Utilization
100	600 sec	1	960 bps	1 Mbps	0.1 %
1 000	600 sec	1	9600 bps	1 Mbps	1.0 %
5 000	600 sec	50	48 kbps	1 Mbps	4.8 %
5 000	300 sec	100	96 kbps	1 Mbps	9.6 %

Table 2: Multicast WiFi Usage by Sleeping Devices

3.3. Low Power WiFi Clients

In order to save their batteries, Low Power (LP) hosts go into radio sleep mode until there is a local need to send a wireless frame. Before going into radio sleep mode, the LP hosts signal to the AP that they are going into sleep; this allows the AP to store unicast and multicast frames destined for those sleeping LP clients. LP clients wake up periodically to listen to the WiFi beacon frames transmitted periodically (default every 100 ms) because this beacon frame contains a bit mask (Traffic Indication Map - TIM) indicating for which STA there is waiting unicast traffic and whether there is multicast traffic waiting. If there is multicast traffic waiting, that ALL LP hosts must stay awake to receive all multicast frames sent immediately after by the AP and process them. If there is a bit indicating that unicast traffic is waiting for a specific LP host, then only this LP host will stay awake to poll the AP later to collect its traffic. The TIM maximum length is 2008 bits and the complete beacon frame is less than 300 bytes long.

The table Table 2 indicates the ration of active/sleeping time for LP hosts when multicast is present. In the absence of multicast traffic, the radio is active only 2.4 % of the time while if there are 50 multicast frames of 300 bytes per second, the radio is active 14.4 % of the time, nearly 6 times more often... with a battery life probably reduced by 6...

Beacon frames/sec	Mcast frames/sec	Mcast frame size (bytes)	Lowest WiFi Rate	Awake time/sec
10	0	300 bytes	1 Mbps	2.4 %
10	5	300 bytes	1 Mbps	3.6 %
10	10	300 bytes	1 Mbps	4.8 %
10	50	300 bytes	1 Mbps	14.4 %

Table 3: Multicast WiFi Impact on Low Power Hosts

3.4. Vendor and Configuration Optimizations

Vendors have noticed the problem and have come with several optimizations such as

- LP hosts not waking up the main processor when they are not member of the multicast group;
- APs no transmitting back over radio received Router Sollication multicast messages;
- ...

AP can also work in 'AP isolation mode' where there is no direct traffic between WiFi clients, this mode has a positive side-effect when a WiFi client transmits a multicast frame as this frame is transmitted at the highest possible rate over the WiFi medium and the AP will not re-transmit if back to all other WiFi clients at the lowest rate.

3.5. Even Unicast NDP is not Optimum

While this is not directly related to the subject of this document, it is worth mentioning anyway as this is important for devices running on battery.

NDP cache needs to be maintained by refreshing the neighbor cache for entries which are in the STALE state. This requires yet another Neighbor Solicitation / Neighbor Advertisement round. Even if the destination IP and MAC addresses are unicast, this traffic is generated and again wakes up mobile devices.

4. Measuring the Amount of IPv6 Multicast

There are basically three ways to measure the amount of IPv6 multicast traffic:

- sniffing the traffic and generating statistics, somehow an overkill:
- exporting IPfix data and doing aggregation on the ff02::/16 link-local multicast prefix
- using SNMP to query on the AP the IP-MIB [RFC4293] with commands such as:
  - `snmpwalk -c private -v 1 udp6:[2001:db8::1] -Ci -m IP-MIB ifDesc`: to get the interface names and index;
  - `snmpwalk -c private -v 1 udp6:[2001:db8::1] -Ci -m IP-MIB ipIfStatsOutTransmits.ipv6`: to get the global transmit counters (i.e. unicast and multicast as there is no broadcast in IPv6);
  - `snmpwalk -c private -v 1 udp6:[2001:db8::1] -Ci -m IP-MIB ipIfStatsOutMcastPkts.ipv6`: to get the multicast packet counter.

5. Acknowledgements

The authors would like to thank Norman Finn, Michel Fontaine, Steve Simlo, Ole Troan, and Stig Venaas for their suggestions and comments.

6. IANA Considerations

This memo includes no request to IANA.

7. Security Considerations

The only security considerations about this document is that by forcing a lot of traffic to be multicast, then, a denial of service (DoS) attack could be mounted on available bandwidth and battery of some network nodes.

8. Informative References

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, February 2006.

[RFC4293] Routhier, S., "Management Information Base for the Internet Protocol (IP)", RFC 4293, April 2006.

[RFC4541] Christensen, M., Kimball, K., and F. Solensky, "Considerations for Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping Switches", RFC 4541, May 2006.

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

[RFC4941] Narten, T., Draves, R., and S. Krishnan, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", RFC 4941, September 2007.

[packet_loss]
Department of Computer Sciences, University of Wisconsin Madison, USA, "Diagnosing Wireless Packet Losses in 802.11: Separating Collision from Weak Signal", <http://pages.cs.wisc.edu/~suman/pubs/diagnose.pdf>.

Authors' Addresses

Eric Vyncke (editor)
Cisco
De Kleetlaan, 6A
Diegem 1831
BE

Phone: +32 2 778 4677
Email: evyncke@cisco.com

Pascal Thubert
Cisco
Batiment D, 45 Allee des Ormes
MOUGINS, PROVENCE-ALPES-COTE D'AZUR 06250
France

Email: pthubert@cisco.com

Eric Levy-Abegnoli
Cisco
Batiment D, 45 Allee des Ormes
MOUGINS, PROVENCE-ALPES-COTE D'AZUR 06250
France

Email: elevyabe@cisco.com

Andrew Yourtchenko
Cisco
De Kleetlaan, 6A
Diegem 1831
BE

Phone: +32 2 704 5494
Email: ayourtch@cisco.com

Network Working Group   A. Yourtchenko
Internet-Draft   cisco
Intended status: Informational   L. Colitti
Expires: August 18, 2014   Google
                                                       February 14, 2014

Reducing Multicast in IPv6 Neighbor Discovery
draft-yourtchenko-colitti-nd-reduce-multicast-00

Abstract

IPv6 Neighbor Discovery protocol makes wide use of multicast traffic, which makes it not energy efficient for the mobile WiFi hosts. This document describes two classes of possible ways to reduce the multicast traffic within IPv6 ND. First, within the boundaries of existing protocols. Second - with what the authors deem to be "minor changes" to the existing protocols.

Status of This Memo

This Internet-Draft will expire on August 18, 2014.

Copyright Notice

Table of Contents

1. Introduction

Wireless networks based on the IEEE 802.11 standard (WiFi) are ubiquitous in today’s life. The multicast/broadcast behavior in these networks has significantly lower performance than unicast in the majority of the cases.

Also, in the current standard and implementations of the 802.11 protocols from the link-layer media standpoint the multicast is the same as broadcast.

The Neighbor Discovery protocol makes substantial use of multicast packets on the assumption that they provide the same or better efficiency compared to unicast packets.

This misalignment results that the nodes on IPv6 networks with the default configuration perform significantly poorer both from the battery life standpoint and the bandwidth efficiency standpoint.

This document presents two groups of measures which reduce the shortcoming:

- The measures which are possible without any changes to the existing standards.

- The measures which require minimal changes to the standards.

Add some text here. You will need to use these references somewhere within the text: [RFC4862] [RFC4861] [RFC6620] [RFC3315]

2. Impact of Multicast Packets in 802.11 Networks

NOTE: much if not all of the subsequent text in this section might need to be transferred to vyncke-6man-mcast-not-efficient-01, which discusses why multicast is not an efficient media in the WiFi environments.

1. Multicast can impact power consumption on hosts if hosts receive multicast packets that are not addressed to them.

2. Excessive use of multicast can reduce the performance of wireless networks.

3. The extra packets are more expensive when they occur with the host not otherwise engaged in using the network.

4. Mobile nodes often have more than one processor and multiple power management states both for the central processing unit and for the WiFi portion (e.g. using only one antenna out of multiple). Often, the battery impact of rejecting a packet in the radio firmware is substantially lower than the impact of passing the packet to the main processor and rejecting it there.

In 802.11 networks, multicast frames towards clients have a greater battery impact than the unicast frames because they are transmitted to all hosts at once, with the AP setting the DTIM bit on the beacon packet to signal to the dozing hosts that the transmission is about to begin.

Thus, if the host were not to wake up right there and then, it would miss the multicast frame. Unicast packets are buffered on the AP and may have a more lenient delivery schedule, which would allow the devices to not have to wake up at every beacon interval (100ms).

The tradeoff between the energy savings and the latency of the multicast delivery may be manipulated by changing the parameter called DTIM interval, which determines how often (every Nth beacon) the AP can send the indication about the multicast traffic to the clients - with the default values being fairly low, usually in the range of one to three.

Increasing these values increases the latency for the multicast packets, therefore changing the DTIM interval beyond the defaults is usually not recommended.

3. Quantifying the use of Multicast in Neighbor Discovery

Normal operation of Neighbor Discovery uses the following multicast packets.

1. Duplicate Address Detection.
   Expected impact: One packet per IPv6 address (a host may be configured to do 2 or more) every time a host joins the network

2. Router Solicitations.
   Expected impact: One packet every time a host joins the network.

3. Router Advertisements.
   Expected impact:

   * One multicast RAs every [RA interval] seconds

   * One solicited RA per host joining the network (if solicited RAs are sent using multicast)

4. Neighbor solicitations. Expected impact: One every time a host talks to a new on-link destination talked to. The response is cached and typically does not expire unless the ND cache is under pressure and subject to garbage collection. Cache entries are refreshed (and possibly deleted) using unicast NUD packets, so cache refreshes do not cause multicast packets to be sent..

With the exception of periodic RAs (and possibly solicited RAs), none of these packets are addressed to all nodes. RS packets are addressed to all routers, and NS packets are addressed to solicited-node multicast groups. Because solicited-node multicast groups contain the last 24 bits of the IPv6 address, in most networks, each solicited-node group will have at most one member.

4. Multicast-limiting measures with no changes in specifications

4.1. On-device robust multicast filtering

The hosts may implement on-device multicast filtering, such that if devices receive multicast packets that are not addressed to them, they will not send the packets to the main CPU but instead remain in a lower sleep state.

It is worth noting that this may require a less deep sleep state than the one required to monitor the TIM in the beacon frames. Also, filtering the packets on the device does not address the inefficiency in spectrum utilisation caused by excessive multicast frames.

4.2. Unicast Solicited Router Advertisements

[RFC4861] in section 6.2.6 already allows to do so via a MAY verb (if the solicitation’s source address is not the unspecified address). This is further weakened by the subsequent qualifier being "but the usual case is to multicast the response to the all-nodes group." As a result of this, a lot of implementations do multicast the solicited RAs, significantly impacting the devices.

To help address this, all router implementations SHOULD have a way to send solicited RAs unicast in the environments which wish to do so.

4.3. Infrastructure-based multicast filtering

Ensure that solicited-node multicasts only go to the specific nodes. This can be implemented either using multicast snooping or by converting multicast packets to unicast packets that are addressed to a subset of the hosts..

The latter can be done in two ways:

- on the 802.11 level alone, preserving the destination within the inner Ethernet frame as multicast
- on the 802.11 and 802.3 levels, as clarified by the [RFC6085]

Some networks track individual device IP addresses for security and tracking reasons, typically by snooping DAD packets or device traffic as described in [RFC6620]

In these networks, the infrastructure is already aware of which IP addresses are mapped to which MAC addresses, and can use this information to selectively unicast neighbor solicitations to the nodes that will be interested in them.

Most wireless networks are infrastructure-based. The 802.11 standard defines that all communications in such networks will happen via the access points. Therefore, the infrastructure has a chance to intelligently filter any multicast packets that are coming from both local (served by the same access point) and remote (located behind the wired infrastructure) hosts or routers, before forwarding them onto the air to their ultimate destination.

4.4. Proxy the Neighbor Discovery protocol on the access point

802.11 standard defines also that all of packets sent from the client to the Access Point (either for the local over-the-air delivery or for forwarding on to the wired side) are acknowledged (even the multicast ones).

With this in mind, in the scenarios like DAD, a proxy ND implementation has inherently a much better chance of working than the "regular" forwarding of the multicast DAD NS (and the return forwarding of the multicast DAD NA in case of DAD collision that was detected).

Therefore, the environments which want to increase the robustness of the DAD, may wish to proxy the ND on behalf of the clients, therefore reducing the overall client-directed multicast traffic (which is unacknowledged) and increasing the robustness against the poor radio conditions.

4.5. Maximized Interval for Periodic RAs

Assuming the solicited RAs are sent unicast, increasing the interval of the periodic RAs is a natural way of further reducing the amount of multicast packets in the air.

The bounding factor is AdvDefaultLifetime, which is limited by the [RFC4861], section 6.1 on the sending side to 9000 seconds.

Thus, to find the "right" value one will have to balance the robustness in the face of higher packet loss on the segment with the energy consumption by the endpoints. Some real-world mid-scale networks (on the order of 10000 hosts within a single /64) successfully used a value of one RA in 1800 seconds.

However, it is impossible to specify the "best" value - everything will depend on the quality of the local WiFi installation and the radio conditions, with the constraint of 9000 seconds currently specified by the standard.

4.6. Increasing the advertised Reachable value

The NUD with the default settings and active traffic will enter the PROBE state as frequently as every ˜30 seconds. [RFC4861] section 7.3.3 defines: "If no response is received after waiting RetransTimer milliseconds after sending the MAX_UNICAST_SOLICIT solicitations, retransmissions cease and the entry SHOULD be deleted. Subsequent traffic to that neighbor will recreate the entry and perform address resolution again."

Short-term connectivity issues at link layer may cause a trigger for the symptoms described in the [RFC7048], therefore triggering the nodes to send multicast neighbor solicitations. However, most of the hosts do not implement at this time the changes suggested there. With the default short timeouts and a wireless environment which forwards multicasts without the filtering, these retransmissions may contribute to further possible failures of NUD in other hosts. In the extreme high density and mobility environments (conferences, stadiums) this may result in avalanche effect and significantly increase the portion of multicast traffic.

Furthermore, an 802.11 segment usually has a single gateway (possibly in a FHRP redundant configuration), therefore making NUD not very useful at all: if that gateway does not function, there is no alternative.

For these kinds of environments it may be useful to significantly increase the REACHABLE_TIME from 30000 milliseconds to 600000 seconds and higher. One possible concern here, however, may be the overflow of the ND table on the gateway, so, again, there is no "best" value suitable for all the networks.

4.7. Clearing the on-link bit in the advertized prefixes

The mobile nodes have generally fairly limited memory, so in the environments where there are thousands of nodes on a single /64, it might be burdensome for them to manage a large neghbor table. Having a lot of hosts with large neighbor tables may mean also a lot of NUD maintenance activity, with the potential for the catastrophic failure of the NUD therefore increasing in the high-density environments.

Clearing the on-link bit in the advertised prefixes causes the hosts to send all the traffic to each other via the default gateway - thus dramatically reducing the size of the neighbor table and the burden of its maintenance on the hosts.

The remaining impact of the link-local addresses still present in the cache can then be mitigated by blocking the direct communications between the hosts at L2, which is a standard feature in the wireless LAN equipment. This operation effectively turns a wireless LAN segment into a collection of point-to-point links between the hosts and the access point, not dissimilar to the operation of private VLANs in the wired LAN case - making the subnet effectively NBMA.

4.8. Explicit creation of state with DHCPv6 address assignment

Turning the WLAN subnet into an NBMA has a consequence that the DAD may no longer work - which may create a problem with the global addresses. Therefore, it may be necessary to transfer the control over the address assignment to a centralized entity.

Also, the 802.11 protocols operate in the unlicensed bands, which means that the radio conditions may vary greatly. The 802.11 LLC protocol itself does have a fairly robust L2 retransmission mechanism for the acknowledged packets (up to 64 retransmissions). However, there still may be times when the radio conditions are so poor that this robustness is not enough. If the network were to use the snooping to maintain the strict policies (e.g. restrict the source addresses of the traffic), merely snooping the ND may not work, and the data-driven recovery mechanisms might be unacceptable.

In these cases one may consider using DHCPv6 as an address assignment mechanism, which would provide the explicit management of state by the client, and the retransmissions required to create the necessary state on the network side without requiring the node to send the data.

4.9. Client link shutdown within the router lifetime expiry

Some nodes after a longer period of time may decide to completely shut down the radio. This will of course result in the best battery usage, but will incur a tradeoff that waking up the client from the network side will be impossible. However, this mode of operation is the only one not using DHCPv6 which may allow complete avoidance of multicast RA packets: if the client never stays awake for longer than the router lifetime, it will not require the multicast RA processing. This optimization is here for completeness of the discussion - since it changes the connectivity of the client.

5. Multicast-limiting measures with small changes in specifications

5.1. Remove the send-side limit on AdvDefaultLifetime of 9000 Seconds

[RFC4861], section 6.1 limits the AdvDefaultLifetime on the sending side to 9000 seconds, while explicitly requiring the receiving side to process all the values up to 65535 (maximum allowed by 16-bit unsigned integer that the AdvDefaultLifetime is).

This artificial limit means a hard limit on the maximum router lifetime that can be specified in the configuration. (The authors tried two router implementations: Cisco IOS and radvd. More information welcome).

This artificial restriction prevents from using very long router advertisement intervals that would otherwise be possible - with the difference being more than 7x!

Additionally, allowing the router lifetime of 65535 seconds, coupled with sufficiently long lifetimes for the prefix, would cover the vast majority of the lifetimes of the devices on the WiFi networks. 65535 seconds is 18.2 hours, and the typical mobile devices might not even stay on the same network for such a long period of time. This would allow to increase the robustness of the network in the face of bad radio conditions causing the high loss of the multicast RAs.

5.2. Explicitly Client-Driven Router Advertisements

We can logically extend the "client link shutdown" in the direction of smaller connectivity loss, and imagine that the client, instead of completely shutting the radio down, would flap its radio link somewhere close to router lifetime expiry, therefore, while acting fully within the standards it will be able to maintain the connectivity during all but very short period of time, without any use of periodic RAs.

It may be interesting to explore a modification of the client behavior such that the "flap time" converges to zero, and eventually allowing the client to initiate a unicast Router Solicitation some time shortly before the router lifetime expires. This will have the result of the client being able to maintain the connectivity without the need of processing any periodic RAs. The advantage of doing so is that the RS-RA exchange will happen at the time convenient for the client sleep schedule - thus allowing to maximize the battery life.

6. Acknowledgements

Thanks to the following people for the very useful discussions. In no particular order: Erik Nordmark, Pascal Thubert, Eric Levy-Abegnoli, Ole Troan, Eric Vyncke, Federico Lovison, Jerome Henry.

7. IANA Considerations

None.

8. Security Considerations

Not discussed in -00.

9. Normative References

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

[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.

[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.

[RFC6085] Gundavelli, S., Townsley, M., Troan, O., and W. Dec, "Address Mapping of IPv6 Multicast Packets on Ethernet", RFC 6085, January 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.

[RFC7048] Nordmark, E. and I. Gashinsky, "Neighbor Unreachability Detection Is Too Impatient", RFC 7048, January 2014.

Authors’ Addresses

Andrew Yourtchenko
cisco
7a de Kleetlaan
Diegem, 1831
Belgium

Phone: +32 2 704 5494
Email: ayourtch@cisco.com

Lorenzo Colitti
Google

Email: lorenzo@google.com