ï»¿

Network Working Group
Internet-Draft
Intended status: Informational
Expires: January 28, 2018

W. Liu
W. Xu
C. Zhou
Huawei Technologies
T. Tsou
Philips Lighting
S. Perreault
Jive Communications
P. Fan

R. Gu
China Mobile
C. Xie
China Telecom
Y. Cheng
China Unicom
July 29, 2017

Gap Analysis for IPv4 Sunset
draft-ietf-sunset4-gapanalysis-09

Abstract

Sunsetting IPv4 refers to the process of turning off IPv4 definitively. It can be seen as the final phase of the transition to IPv6. This memo enumerates difficulties arising when sunsetting IPv4, and identifies the gaps requiring additional work.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

The final phase of the transition to IPv6 is the sunset of IPv4, that is turning off IPv4 definitively on the attached networks and on the upstream networks.

Some current implementation behavior makes it hard to sunset IPv4. Additionally, some new features could be added to IPv4 to make its sunsetting easier. This document analyzes the current situation and proposes new work in this area.

The decision about when to turn off IPv4 is out of scope. This document merely attempts to enumerate the issues one might encounter if that decision is made.

2. Related Work

[RFC3789], [RFC3790],[RFC3791], [RFC3792], [RFC3793], [RFC3794], [RFC3795] and [RFC3796] contain surveys of IETF protocols with their IPv4 dependencies.

Additionally, although reviews in RFCs 3789-3796 ensured that IETF standards then in use could support IPv6, no IETF-wide effort has been undertaken to ensure that the issues identified in those drafts are all addressed, nor to ensure that standards written after RFC3100 (where the previous review efforts stopped) function properly on IPv6-only networks.

The IETF needs to ensure that existing standards and protocols have been actively reviewed, and any parity gaps either identified so that they can be fixed, or documented as unnecessary to address because it is unused or superseded by other features.

First, the IETF must review RFCs 3789-3796 to ensure that any gaps in specifications identified in these documents and still in active use have been updated as necessary to enable operation in IPv6-only environments (or if no longer in use, are declared historic).

Second, the IETF must review documents written after the existing review stopped (according to RFC 3790, this review stopped with approximately RFC 3100) to identify specifications where IPv6-only operation is not possible, and update them as necessary and appropriate, or document why an identified gap is not an issue i.e. not necessary for functional parity with IPv4.

This document does not recommend excluding Informational and BCP RFCs as the previous effort did, due to changes in the way that these documents are used and their relative importance in the RFC Series. Instead, any documents that are still active (i.e. not declared historic or obsolete) and the product of IETF consensus (i.e. not a product of the ISE Series) should be included. In addition, the reviews undertaken by RFCs 3789-3796 were looking for "IPv4 dependency" or "usage of IPv4 addresses in standards". This document recommends a slightly more specific set of criteria for review. Reviews should include:

- o Consideration of whether the specification can operate in an environment without IPv4.

- o Guidance on the use of 32-bit identifiers that are commonly populated by IPv4 addresses.

- o Consideration of protocols on which specifications depend or interact, to identify indirect dependencies on IPv4.

- o Consideration of how to transit from an IPv4 environment to an IPv6 environment.

3. Remotely Disabling IPv4

3.1. Indicating that IPv4 connectivity is unavailable

PROBLEM 1: When an IPv4 node boots and requests an IPv4 address (e.g., using DHCP), it typically interprets the absence of a response as a failure condition even when it is not.

PROBLEM 2: Home router devices often identify themselves as default routers in DHCP responses that they send to requests coming from the LAN, even in the absence of IPv4 connectivity on the WAN.

3.2. Disabling IPv4 in the LAN

PROBLEM 3: IPv4-enabled hosts inside an IPv6-only LAN can auto-configure IPv4 addresses [RFC3927] and enable various protocols over IPv4 such as mDNS [RFC6762] and LLMNR [RFC4795]. This can be undesirable for operational or security reasons, since in the absence of IPv4, no monitoring or logging of IPv4 will be in place.

PROBLEM 4: IPv4 can be completely disabled on a link by filtering it on the L2 switching device. However, this may not be possible in all cases or may be too complex to deploy. For example, an ISP is often not able to control the L2 switching device in the subscriber home network.

PROBLEM 5: A host with only Link-Local IPv4 addresses will "ARP for everything", as described in Section 2.6.2 of [RFC3927]. Applications running on such a host connected to an IPv6-only network will believe that IPv4 connectivity is available, resulting in various bad or sub-optimal behavior patterns. See [I-D.yourtchenko-ipv6-disable-ipv4-proxyarp] for further analysis.

Some of these problems were described in [RFC2563], which standardized a DHCP option to disable IPv4 address auto-configuration. However, using this option requires running an IPv4 DHCP server, which is contrary to the goal of IPv4 sunsetting.

4. Client Connection Establishment Behavior

PROBLEM 6: Happy Eyeballs [RFC6555] refers to multiple approaches to dual-stack client implementations that try to reduce connection setup delays by trying both IPv4 and IPv6 paths simultaneously. Some implementations introduce delays which provide an advantage to IPv6, while others do not [Huston2012]. The latter will pick the fastest path, no matter whether it is over IPv4 or IPv6, directing more traffic over IPv4 than the other kind of implementations. This can prove problematic in the context of IPv4 sunsetting, especially for Carrier-Grade NAT phasing out because CGN does not add significant latency that would make the IPv6 path more preferable. Traffic will therefore continue using the CGN path unless other network conditions change.

PROBLEM 7: getaddrinfo() [RFC3493] sends DNS queries for both A and AAAA records regardless of the state of IPv4 or IPv6 availability. The AI_ADDRCONFIG flag can be used to change this behavior, but it relies on programmers using the getaddrinfo() function to always pass this flag to the function. The current situation is that in an IPv6-only environment, many useless A queries are made.

5. Disabling IPv4 in Operating System and Applications

It is possible to completely remove IPv4 support from an operating system as has been shown by the work of Bjoern Zeeb on FreeBSD. [Zeeb] Removing IPv4 support in the kernel revealed many IPv4 dependencies in libraries and applications.

PROBLEM 8: Completely disabling IPv4 at runtime often reveals implementation bugs. Hard-coded dependencies on IPv4 abound, such as on the 127.0.0.1 address assigned to the loopback interface, and legacy IPv4-only APIs are widely used by applications. It is hard for the administrators and users to know what applications running on the operating system have implementation problems of IPv4 dependency. It is therefore often operationally impossible to completely disable IPv4 on individual nodes.

PROBLEM 9: In an IPv6-only world, legacy IPv4 code in operating systems and applications incurs a maintenance overhead and can present security risks.

6. On-Demand Provisioning of IPv4 Addresses

As IPv6 usage climbs, the usefulness of IPv4 addresses to subscribers will become smaller. This could be exploited by an ISP to save IPv4 addresses by provisioning them on-demand to subscribers and reclaiming them when they are no longer used. This idea is described in [I-D.fleischhauer-ipv4-addr-saving] and [BBF.TR242] for the context of PPP sessions. In these scenarios, the home router is responsible for requesting and releasing IPv4 addresses, based on snooping the traffic generated by the hosts in the LAN, which are still dual-stack and unaware that their traffic is being snooped.

As described in TR-092 and TR-187, NAS(e.g., BRAS, BNG) stores pools of IPv4 and IPv6 addresses, which are used for DHCP distribution to the hosts in home network. IPv4 and IPv6 addresses of hosts can be dynamic assignment from a pool of IPv4 and IPv6 prefixes in NAS.

As the IPv4 sunsets, the number of IPv4 hosts is reduced, therefore the IPv4 address resource in NAS needs to be reduced too. These reduced IPv4 addresses will be reclaimed by the address management system (NMS, controller, IPAM, etc.). At the same time, as the number of IPv6 hosts increases, NAS need incrementally increase the number of IPv6 address resource. The increased IPv6 address resource can be assigned by the address management system, which makes the transition more smoothly by dynamically adding / releasing IP address resources in NAS. In modern network systems, protocols such as NETCONF / RESTCONF / RADIUS can be used for this process. With NETCONF, NAS acts as NETCONF server with the opening port to listen for the client connection, while the address management system as a netconf client that connects and processes IP address request from NAS.

PROBLEM 10: Dual-stack hosts that implement Happy-Eyeballs [RFC6555] will generate both IPv4 and IPv6 traffic even if the algorithm end up chooosing IPv6. This means that an IPv4 address will always be requested by the home router, which defeats the purpose of on-demand provisioning.

PROBLEM 11: Many operating systems periodically perform some kind of network connectivity check as long as an interface is up. Similarly, applications often send keep-alive traffic continuously. This permanent "background noise" will prevent an IPv4 address from being released by the home router.

PROBLEM 12: Hosts in the LAN have no knowledge that IPv4 is available to them on-demand only. If they had explicit knowledge of this fact, they could tune their behaviour so as to be more conservative in their use of IPv4.

PROBLEM 13: This mechanism is only being proposed for PPP even though it could apply to other provisioning protocols (e.g., DHCP).

PROBLEM 14: When the number of IPv4 hosts connected to NAS is reduced, the NAS releases the IPv4 address resource and the NAS requests more IPv6 address resource for it to serve hosts transitting from IPv4 to IPv6.

7. IPv4 Address Literals

IPv4 addresses are often used as resource locators. For example, it is common to encounter URLs containing IPv4 address literals on web

sites [I-D.wing-behave-http-ip-address-literals]. IPv4 address literals may be published on media other than web sites, and may appear in various forms other than URLs. For the operating systems which exhibit the behavior described in [I-D.yourtchenko-ipv6-disable-ipv4-proxyarp], this also means an increase in the broadcast ARP traffic, which may be undesirable.

PROBLEM 15: IPv6-only hosts are unable to access resources identified by IPv4 address literals.

8. Managing Router Identifiers

IPv4 addresses are often conventionally chosen to number a router ID, which is used to identify a system running a specific protocol. The common practice of tying an ID to an IPv4 address gives much operational convenience. A human-readable ID is easy for network operators to deal with, and it can be auto-configured, saving the work of planning and assignment. It is also helpful to quickly perform diagnosis and troubleshooting, and easy to identify the availability and location of the identified router.

PROBLEM 16: In an IPv6 only network, there is no IP address that can be directly used to number a router ID. IDs have to be planned individually to meet the uniqueness requirement. Tying the ID directly to an IP address which yields human-friendly, auto-configured ID that helps with troubleshooting is not possible.

9. IANA Considerations

None.

10. Security Considerations

It is believed that none of the problems identified in this draft are security issues.

11. Acknowledgements

Thanks in particular to Andrew Yourtchenko, Jordi Palet Martinez, Lee Howard, Nejc Skoberne, and Wes George for their thorough reviews and comments.

Special thanks to Marc Blanchet who was the driving force behind this work and to Jean-Philippe Dionne who helped with the initial version of this document.

12. Informative References

[BBF.TR242]
Broadband Forum, "TR-242: IPv6 Transition Mechanisms for Broadband Networks", August 2012.

[Huston2012]
Huston, G. and G. Michaelson, "RIPE 64: Analysing Dual Stack Behaviour and IPv6 Quality", April 2012.

[I-D.fleischhauer-ipv4-addr-saving]
Fleischhauer, K. and O. Bonness, "On demand IPv4 address provisioning in Dual-Stack PPP deployment scenarios", draft-fleischhauer-ipv4-addr-saving-05 (work in progress), September 2013.

[I-D.wing-behave-http-ip-address-literals]
Wing, D., "Coping with IP Address Literals in HTTP URIs with IPv6/IPv4 Translators", draft-wing-behave-http-ip-address-literals-02 (work in progress), March 2010.

[I-D.yourtchenko-ipv6-disable-ipv4-proxyarp]
Yourtchenko, A. and O. Owen, "Disable "Proxy ARP for Everything" on IPv4 link-local in the presence of IPv6 global address", draft-yourtchenko-ipv6-disable-ipv4-proxyarp-00 (work in progress), May 2013.

[RFC2563] Troll, R., "DHCP Option to Disable Stateless Auto-Configuration in IPv4 Clients", RFC 2563, May 1999.

[RFC3493] Gilligan, R., Thomson, S., Bound, J., McCann, J., and W. Stevens, "Basic Socket Interface Extensions for IPv6", RFC 3493, February 2003.

[RFC3789] Nesser, P. and A. Bergstrom, "Introduction to the Survey of IPv4 Addresses in Currently Deployed IETF Standards Track and Experimental Documents", RFC 3789, June 2004.

[RFC3790] Mickles, C. and P. Nesser, "Survey of IPv4 Addresses in Currently Deployed IETF Internet Area Standards Track and Experimental Documents", RFC 3790, June 2004.

[RFC3791] Olvera, C. and P. Nesser, "Survey of IPv4 Addresses in Currently Deployed IETF Routing Area Standards Track and Experimental Documents", RFC 3791, June 2004.

[RFC3792] Nesser, P. and A. Bergstrom, "Survey of IPv4 Addresses in Currently Deployed IETF Security Area Standards Track and Experimental Documents", RFC 3792, June 2004.

[RFC3793] Nesser, P. and A. Bergstrom, "Survey of IPv4 Addresses in Currently Deployed IETF Sub-IP Area Standards Track and Experimental Documents", RFC 3793, June 2004.

[RFC3794] Nesser, P. and A. Bergstrom, "Survey of IPv4 Addresses in Currently Deployed IETF Transport Area Standards Track and Experimental Documents", RFC 3794, June 2004.

[RFC3795] Sofia, R. and P. Nesser, "Survey of IPv4 Addresses in Currently Deployed IETF Application Area Standards Track and Experimental Documents", RFC 3795, June 2004.

[RFC3796] Nesser, P. and A. Bergstrom, "Survey of IPv4 Addresses in Currently Deployed IETF Operations & Management Area Standards Track and Experimental Documents", RFC 3796, June 2004.

[RFC3927] Cheshire, S., Aboba, B., and E. Guttman, "Dynamic Configuration of IPv4 Link-Local Addresses", RFC 3927, May 2005.

[RFC4795] Aboba, B., Thaler, D., and L. Esibov, "Link-local Multicast Name Resolution (LLMNR)", RFC 4795, January 2007.

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

[RFC6762] Cheshire, S. and M. Krochmal, "Multicast DNS", RFC 6762, February 2013.

[Zeeb] "FreeBSD Snapshots without IPv4 support", <http://wiki.freebsd.org/IPv6Only>.

Annex A. Solution Ideas

A.1. Remotely Disabling IPv4

A.1.1. Indicating that IPv4 connectivity is unavailable

One way to address these issues is to send a signal to a dual-stack node that IPv4 connectivity is unavailable. Given that IPv4 shall be off, the message must be delivered through IPv6.

A.1.2. Disabling IPv4 in the LAN

One way to address these issues is to send a signal to a dual-stack node that auto-configuration of IPv4 addresses is undesirable, or that direct IPv4 communication between nodes on the same link should not take place.

A signalling protocol equivalent to the one from [RFC2563] but over IPv6 is necessary, using either Router Advertisements or DHCPv6.

Furthermore, it could be useful to have L2 switches snoop this signalling and automatically start filtering IPv4 traffic as a consequence.

Finally, it could be useful to publish guidelines on how to safely block IPv4 on an L2 switch.

A.2. Client Connection Establishment Behavior

Recommendations on client connection establishment behavior that would facilitate IPv4 sunsetting would be appropriate.

Happy Eyeballs timers and related parameters should get gradually increased, so even if IPv6 is "slower" than IPv4, IPv6 gains preference anyway.

A.3. Disabling IPv4 in Operating System and Applications

It would be useful for the IETF to provide guidelines to programmers on how to avoid creating dependencies on IPv4, how to discover existing dependencies, and how to eliminate them. It would be useful if operating systems provide functions for users to see what applications uses legacy IPv4-only APIs, so they can know it better whether they can turn off IPv4 completely. Having programs and operating systems that behave well in an IPv6-only environment is a prerequisite for IPv4 sunsetting.

A.4. On-Demand Provisioning of IPv4 Address

As the sunset of IPv4 in NAS, parts of hosts no longer need IPv4 address. IPv4 address resources in NAS appears surplus, NAS should obtain the unoccupied IPv4 address, generate a request and send it to the address management system to release those IPv4 address resource. Meanwhile, NAS needs more IPv6 address resources for the host transiting from IPv4 to IPv6. NAS judges whether the usage status of the IPv6 address resource satisfies certain condition, and the condition can be IPv6 address utilization ritio. If the IPv6 address utilization ratio is too high, the NAS generates a resource request containing IPv6 addresses information that needs to be applied and sends it to the address management system. When the address management system receives the IPv6 address resource request, it allocates IPv6 address pool from its assignable IPv6 address resource according to the information of the resource request, then it sends a response message with the information of allocated IPv6 address pool for this NAS to the NAS. Then the NAS receives the response and gets the information of allocated IPv6 address pool.

A.5. Managing Router Identifiers

Router IDs can be manually planned, possibly with some hierarchy or design rule, or can be created automatically. A simple way of automatic creation is to generate pseudo-random numbers, and one can use another source of data such as the clock time at boot or configuration time to provide additional entropy during the generation of unique IDs. Another way is to hash an IPv6 address down to a value as ID. The hash algorithm is supposed to be known and the same across the domain. Since typically the number of routers in a domain is far smaller than the value range of IDs, the hashed IDs are hardly likely to conflict with each other, as long as the hash algorithm is not designed too badly. It is necessary to be able to override the automatically created value, and desirable if the mechanism is provided by the system implementation.

If the ID is created from IPv6 address, e.g. by hashing from an IPv6 address, then naturally it has relationship with the address. If the ID is created regardless of IP address, one way to build association with IPv6 address is to embed the ID into an IPv6 address that is to be configured on the router, e.g. use a /96 IPv6 prefix and append it with a 32-bit long ID. One can also use some record keeping mechanisms, e.g. text file, DNS or other provisioning system like network management system to manage the IDs and mapping relations

with IPv6 addresses, though extra record keeping does introduce additional work.

Authors' Addresses

Will(Shucheng) Liu
Huawei Technologies
Bantian, Longgang District
Shenzhen 518129
China

Email: liushucheng@huawei.com

Weiping Xu
Huawei Technologies
Bantian, Longgang District
Shenzhen 518129
China

Email: xuweiping@huawei.com

Cathy Zhou
Huawei Technologies
Bantian, Longgang District
Shenzhen 518129
China

Email: cathy.zhou@huawei.com

Tina Tsou
Philips Lighting
United States of America

Email: tina.tsou@philips.com

Simon Perreault
Jive Communications
Quebec, QC
Canada

Email: sperreault@jive.com

Peng Fan
Beijing
China

Email: fanp08@gmail.com

Rong Gu
China Mobile
32 Xuanwumen West Ave, Xicheng District
Beijing 100053
China

Email: gurong_cmcc@outlook.com

Chongfeng Xie
China Telecom
China Telecom Beijing Information Science&Technology Innovation Park
Beiqijia Town Changping District, Beijing 102209,
China

Email: xiechf.bri@chinatelecom.cn

Ying Cheng
China Unicom
No.21 Financial Street, XiCheng District
Beijing 100033
China

Email: chengying10@chinaunicom.cn

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: June 7, 2015

S. Perreault
Jive Communications
W. George
Time Warner Cable
T. Tsou
Huawei Technologies (USA)
T. Yang
L. Li
China Mobile
JF. Tremblay
Viagenie

December 4, 2014

Turning off IPv4 Using DHCPv6 or Router Advertisements
draft-ietf-sunset4-noipv4-01

Abstract

This memo defines a new DHCPv6 option and a new Router Advertisement option to inform a dual-stack host or router that IPv4 can be turned off.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

When a dual-stack host makes a DHCPv4 request, it typically interprets the absence of a response as a failure condition. This may cause operational problems when deploying an IPv6-only network. Providing a way to inform hosts and routers that IPv4 is not available would prevent such problems and allow for smoother deployments.

One situation where problems arise is with a dual-stack home router provisioned with an IPv6-only WAN connection. It typically assigns an IPv4 address to its LAN interface, starts services on that interface and hands out IPv4 addresses to clients on the LAN by answering DHCPv4 requests. This is done unconditionally, without

taking the status of the IPv4 connectivity on the WAN interface into account. Hosts on the LAN install a default route pointing to the router and behave as if IPv4 connectivity was available. IPv4 packets destined to the Internet get dropped at the router and timeouts happen. The end result is that IPv4 remains fully active on the LAN and on the router itself even if it would be desirable to turn it off, especially for applications that do not implement Happy Eyeballs [RFC6555].

Another situation relates to the load on DHCPv4 servers and relays. In large dual-stack network (LAN, WLAN), thousands of hosts, including mobile phones, may generate a significant amount of trafic by attempting to contact a DHCP server. If the servers and relays are configured in IPv6-only, the dual-stack or IPv4-only clients will broadcast DHCPDISCOVER messages endlessly, creating a DDOS-like attack on the network. This scenario has also been briefly described for DHCPv6 in [RFC7083]. Although DHCP mandates a exponential backoff, it is limited to 64 seconds, which may still generate significant traffic (see section 4.1 of [RFC2131]). Various operating systems also implement the backoff algorithms in different ways, or not at all, with different limit values. Some test results for a few popular operating systems are available in appendix.

A new mechanism is needed to indicate the absence of IPv4 connectivity. Considering the end goal is turn off all IPv4 connectivity, the chosen mechanism should be transported over IPv6. Therefore, this document introduce a new DHCPv6 [RFC3315] option and a new Router Advertisement (RA) [RFC4861] option for the purpose of explicitly indicating to the host that IPv4 connectivity is unavailable.

2. Terminology

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

The following terms are also used in this document:

Upstream Interface: An interface on which the No-IPv4 option is received over either DHCPv6 or RA.

3. Problems Being Addressed

3.1. Load on DHCPv4 Server and Relay

When a DHCPv4 server or relay is present but intentionally does not react to DHCPDISCOVERs, the aggregated traffic generated by a large number of dual-stack hosts can represent a significant bandwidth load. This scenario is encountered with an ISP serving multiple types of subscribers where some a provisioned for IP4 service and others are not. It might not be feasible for operational reasons to block the useless requests before they reach the DHCPv4 servers, for example if the DHCPv4 servers themselves are the only ones with the knowledge of which nodes should or should not get an IPv4 address.

3.2. Bandwidth Consumption

In addition to the useless load on the DHCPv4 servers, the above scenario could also consume a significant amount of bandwidth, especially if the aggregated traffic from many clients goes through a low-bandwidth link or through a wireless link.

3.3. Power Inefficiency

A dual-stack node that does not get a DHCPv4 response will usually continue retransmitting forever. Therefore, only providing IPv6 on a link will cause the node to needlessly wake up periodically and transmit a few packets. For example, the popular DHCPv4 client implementation by ISC wakes up every 5 minutes by default and tries to contact a DHCPv4 server for 60 seconds. With this configuration, a node will not be able to sleep 20% of the time.

3.4. IPv4 Only Applications

In many cases, IPv4-only applications such as Skype use an autoconfigured IPv4 Link-Local Addresses (LLA) to send IPv4 packets on the LAN. In an IPv6-only environment, this behavior may waste a significant amount of bandwidth.

4. Design Considerations

4.1. DHCPv6 vs DHCPv4

NOTE: This section will be removed before publication as an RFC.

This document describes a new DHCPv6 option to turn off IPv4. An equivalent option could conceivably be created for DHCPv4. The pros and cons are discussed below. Arguments with a + sign argue for a DHCPv4 option, arguments with a - sign argue against.

+ Devices that don’t speak IPv6 won’t be listening for a "turn off IPv4" code, and therefore won’t stop trying to establish IPv4 connectivity.

  - Devices that haven’t been updated to speak IPv6 likely won’t recognize a new DHCPv4 code telling them that IPv4 isn’t supported.

    + However, it’s easier to implement something that turns off the IP stack than implement IPv6.

- Devices that don’t speak IPv6 that are still active on the network mean that either IPv4 can’t/shouldn’t be turned off yet, or IPv4 local connectivity should be maintained to retain local services, even if global IPv4 connectivity is not necessary (think local LAN DLNA streaming, etc).

- When the goal is to turn off IPv4, having to maintain and operate an IPv4 infrastructure (routing, ACLs, etc.) just to be able to send negative responses to DHCPv4 requests is not productive. Having the option transported in IPv6 allows the ISP to focus on operating an IPv6-only network.

  + However, a full IPv4 infrastructure would not be necessary in many cases. The local router could contain a very restricted DHCPv4 server function whose only purpose would be to reply with the No-IPv4 option. No IPv4 traffic would have to be carried to a distant DHCPv4 server. Note however that this may not be operationally feasible in some situations.

- Turning IPv4 off using an IPv4-transported signal means that there is no way to go back. Once the DHCPv4 option has been accepted by the DHCPv4 client, IPv4 can no longer be turned on remotely (rebooting the client still works). Configurations change, mistakes happen, and so it is necessary to have a way to turn IPv4 back on. With a DHCPv6 option, IPv4 can be turned back on as soon as the client makes a new DHCPv6 request, which can be the next scheduled one or can be triggered immediately with a Reconfigure message.

The authors conclude that a DHCPv6 option is clearly necessary, whereas the need for a DHCPv4 option is not as obvious. More feedback on this topic would be appreciated.

4.2. DHCPv6 vs RA

Both DHCPv6 and RA-based solutions are presented in this draft. It is expected that the working group will decide whether both solutions, only one, or none are desirable.

5. The No-IPv4 DHCPv6 Option

The No-IPv4 DHCPv6 option is used to signal the unavailability of IPv4 connectivity.

5.1. DHCPv6 Wire Format

The format of the DHCPv6 No-IPv4 option is:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         OPTION_NO_IPV4        |          option-len           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    v4-level   |
+-+-+-+-+-+-+-+-+
```

option-code OPTION_NO_IPV4 (TBD).

option-len 1.

v4-level Level of IPv4 functionality.

The DHCPv6 client MUST place the OPTION_NO_IPV4 option code in the Option Request Option ([RFC3315] section 22.7). Servers MAY include the option in responses (if they have been so configured). Servers MAY also place the OPTION_NO_IPV4 option code in an Option Request Option contained in a Reconfigure message.

5.2. RA Wire Format

The format of the RA No-IPv4 option is:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |    Length     |   v4-level    |    Reserved   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Reserved                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type TBD

Length 1.

v4-level Level of IPv4 functionality.

Reserved These fields are unused. They MUST be initialized to zero by the sender and MUST be ignored by the receiver.

5.3. Semantics

The option applies to the link on which it is received. It is used to indicate to the client that it should disable some or all of its IPv4 functionality. What should be disabled depends on the value of v4-level.

v4-level can take the following values:

0 - IPv4 fully enabled: This is equivalent to the absence of the No-IPv4 option. It is included here so that a DHCPv6 server can explicitly re-enable IPv4 access by including it in a Reply message following a Reconfigure, or similarly by a router in a spontaneous Router Advertisement.

1 - No IPv4 upstream: Any kind of IPv4 connectivity is unavailable on the link on which the option is received. Therefore, any attempts to provision IPv4 by the host or to use IPv4 in any fashion, on that link, will be useless. IPv4 MAY be dropped, blocked, or otherwise ignored on that link.

Upon reception of the No-IPv4 option with value 1, the following IPv4 functionality MUST be disabled on the Upstream Interface:

A. IPv4 addresses MUST NOT be assigned.

B. Currently-assigned IPv4 addresses MUST be unassigned.

C. Dynamic configuration of link-local IPv4 addresses [RFC3927] MUST be disabled.

D. IPv4, ICMPv4, or ARP packets MUST NOT be sent.

E. IPv4, ICMPv4, or ARP packets received MUST be ignored.

F. DNS A queries MUST NOT be sent, even transported over IPv6.

2 - No IPv4 upstream, local IPv4 restricted: Same semantics as value 1, with the following additions:

If all DHCPv6- or RA-configured interfaces receive the No-IPv4 option with a mix of values 1, 2, and 3 (but not exclusively 3), and no other interface provides IPv4 connectivity to the Internet, IPv4 is partially shut down, leaving only local connectivity active. On the Upstream Interface, IPv4 MUST be shut down as listed above. On other interfaces, IPv4 addresses MUST NOT be assigned except for the following:

* Loopback (127.0.0.0/8)

* Link Local (169.254.0.0/16) [RFC3927]

* Private-Use (10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16) [RFC1918]

3 - No IPv4 at all: This is intended to be a stricter version of the above.

The host or router receiving this option MUST disable IPv4 functionality on the Upstream Interface in the same way as for value 1 or 2.

If all DHCPv6 or RA-configured interfaces received the No-IPv4 option with value 3, and no other interface provides IPv4 connectivity to the Internet, IPv4 is completely shut down. In particular:

A. IPv4 address MUST NOT be assigned to any interface.

B. Currently-assigned IPv4 addresses MUST be unassigned.

C. Dynamic configuration of link-local IPv4 addresses [RFC3927] MUST be disabled.

D. IPv4, ICMPv4, or ARP packets MUST NOT be sent on any interface.

E. IPv4, ICMPv4, or ARP packets received on any interface MUST be ignored.

F. In the above, "any interface" includes loopback interfaces. In particular, the 127.0.0.1 special address MUST be removed.

G. Server programs listening on IPv4 addresses (e.g., a DHCPv4 server) MAY be shut down.

H. DNS A queries MUST NOT be sent, even transported over IPv6.

I. If the host or router also runs a DHCPv6 server, it SHOULD include the No-IPv4 option with value 2 in DHCPv6 responses it sends to clients that request it, unless prohibited by local policy. If it currently has active clients, it SHOULD send a Reconfigure to each of them with the OPTION_NO_IPV4 included in the Option Request Option.

J. If the router sends Router Advertisement, it SHOULD include the No-IPv4 option with value 2 in RA messages it sends, unless prohibited by local policy. It SHOULD also send RAs immediately so that the changes take effect for all current hosts.

The intent is to remove all traces of IPv4 activity. Once the No-IPv4 option with value 3 is activated, the network stack should behave as if IPv4 functionality had never been present. For example, a modular kernel implementation could accomplish the above by unloading the IPv4 kernel module at run time.

5.4. Example

A dual-stack home gateway is set up with a single WAN uplink and is configured to use DHCPv4 and DHCPv6 to automatically obtain IPv4 and IPv6 connectivity. On the LAN side, it has one link with multiple hosts.

When it boots, the router assigns 192.168.1.1/24 to its LAN interfaces and starts a DHCPv4 server listening on it. It hands out addresses 191.168.1.100-199 to clients. It also starts an IPv6 Router Advertisement daemon as well as a stateless DHCPv6 server, also listening on the LAN interfaces.

On the WAN side, it starts two provisioning procedures in parallel: one for IPv4 and one for IPv6.

At this point, the ISP does not know if the router supports IPv6-only operation. Therefore, by default, the ISP responds to DHCPv4 requests as usual.

As part of the IPv6 provisioning procedure, the router sends a DHCPv6 request containing OPTION_NO_IPV4 in an Option Request Option. The ISP's DHCPv6 server's reply includes the No-IPv4 option with value 3. When this procedure finishes, the ISP has determined that this customer will run in IPv6-only mode and starts dropping all IPv4 packets at the first hop. If an IPv4 address was assigned, it is reclaimed, and possibly reassigned to another subscriber.

The home router aborts the IPv4 provisioning procedure (if it is still running) and deactivates all IPv4 functionality. It shuts down its DHCPv4 server. It also configures its own stateless DHCPv6 server to send the No-IPv4 option to clients that request it. (JFT: What happens if the timer below is not implemented and IPv4 completes before IPv6? Maybe we could recommend to run IPv6 provisioning first when OPTION_NO_IPV4 is supported.)

As an optimization, the router could delay setting up IPv4 by a few seconds (10 seconds seems reasonable). If the IPv6 procedure completes with the No-IPv4 option during that time, IPv4 will never have been set up and the router will operate in pure IPv6-only mode from the start.

6. Security Considerations

One security concern is that an attacker could use the No-IPv4 option to deny IPv4 access to a victim. However, unprotected vanilla DHCP can already be exploited to cause such a denial of service ([RFC2131] section 7).

TO BE COMPLETED

7. IANA Considerations

IANA is requested to assign value TBD with description OPTION_NO_IPV4 in the "DHCP Option Codes" table which is part of the dhcpv6-parameters registry [1].

IANA is requested to assign value TBD with description "No-IPv4 Option" in the IPv6 Neighbor Discovery Option Formats table which is part of the icmpv6-parameters registry.

8. Acknowledgements

Thanks in particular to Marc Blanchet who was the driving force behind this work.

Rajiv Asati contributed section Section 3.4.

9. References

9.1. Normative References

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

[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.

[RFC3927] Cheshire, S., Aboba, B., and E. Guttman, "Dynamic Configuration of IPv4 Link-Local Addresses", RFC 3927, May 2005.

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

9.2. Informative References

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, March 1997.

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

[RFC7083] Droms, R., "Modification to Default Values of SOL_MAX_RT and INF_MAX_RT", RFC 7083, November 2013.

9.3. URIs

[1] http://www.iana.org/assignments/dhcpv6-parameters

Appendix A. Test Results of Terminals Behavior

In RFC3315 [RFC3315, DHCPv6], SOL_MAX_RT is defined in DHCPv6 to prevent the frequently requesting of clients, which reduces the aggregated traffic. But in RFC2131 [RFC2131, DHCPv4], there are not corresponding IPv4 definitions or options for client’s behavior if the server does not respond for the Discover messages.

In fact, most of the terminals creat backoff algorithms to help them retransmit DHCPDISCOVER message in different frequency according to

their state machine. The same point of almost all the verious Operating Systems is that they could not stop DHCPDISCOVER requests to the server. And that will cause DDoS-Like attack to the server and bandwidth consumption in the link.

We test some of the most popular terminals' OS in WLAN, the results are illuminated as below.

DHCP Discovery Packages Time Table

No	Windows7 Time	Windows7 Time offset	Windows XP Time	Windows XP Time offset	IOS_5.0.1 Time	IOS_5.0.1 Time offset	Android_2.3.7 Time	Android_2.3.7 Time offset	Symbian_S60 Time	Symbian_S60 Time offset
1	0		0		0.1		7.8		0	
2	3.9	3.9	0.1	0.1	1.4	1.3	10.3	2.5	2	2
3	13.3	9.4	4.1	4	3.8	2.4	17.9	7.6	6	4
4	30.5	17.2	12.1	8	7.9	4.1	33.9	16	8	2
5	62.8	32.3	29.1	17	16.3	8.4	36.5	2.6	12	4
6	65.9	3.1	64.9	35.8	24.9	8.6	reconnect		14	2
7	74.9	9	68.9	4	33.4	8.5	56.6	20.1	18	4
8	92.1	17.2	77.9	9	42.2	8.8	60.2	3.6	20	2
9	395.2	303.1	93.9	16	50.8	8.6	68.4	8.2	24	4
10	399.1	3.9	433.9	340	59.1	8.3	84.8	16.4	26	2
11	407.1	8	438.9	5	127.3	68.2	86.7	1.9	30.1	4.1
12	423.4	16.3	447.9	9	128.9	1.6	reconnect		32.1	2
13	455.4	32	464.9	17	131.1	2.2	106.7	20	36.1	4
14	460.4	5	794.9	330	135.1	4	111.4	4.7	38.1	2
15	467.4	7	799.9	5	143.4	8.3	120.6	9.2	42.1	4
16	483.4	16	808.9	9	151.7	8.3	134.9	14.3	44.1	2
17	842.9	359.5	824.9	16	160.4	8.7	136.8	1.9	48.2	4.1
18	846.9	4	1141.9	317	168.8	8.4	reconnect		50.2	2

Figure:Terminals DHCPDISCOVER requests when Server's DHCPv4 module is down

In this figure:

For Windows7, it seems to initiate 8 times DHCPDISCOVER requests in about 300s interval.

For WindowsXP, firstly it launches 9 times DHCPDISCOVER messages, but after that it cannot get any response from the server, then it

initiates 5 times requests in one cycle in around 330s intervals, and never stop.

For IOS5.0.1, it seems like WindowsXP. There are 10 times attempts in one cycle, and the interval is about 68s.

Symbian_S60 uses the simplest backoff method, it launches DISCOVER in every 2 or 4 seconds.

Android2.3.7 is the only Operating System which can stop DISCOVER request by disconnect its wireless connection. It reboot wireless and dhcp connection every 20 seconds.

Authors' Addresses

Simon Perreault
Jive Communications
Quebec, QC
Canada

Email: sperreault@jive.com

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

Email: wesley.george@twcable.com

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

Phone: +1 408 330 4424
Email: tina.tsou.zouting@huawei.com

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

Email: yangtianle@chinamobile.com

Li Lianyuan
China Mobile
32, Xuanwumenxi Ave.
Xicheng District, Beijing 100053
China

Email: lilianyuan@chinamobile.com

Jean-Francois Tremblay
Viagenie
246 Aberdeen
Quebec, QC G1R 2E1
Canada

Phone: +1 418 656 9254
Email: jean-francois.tremblay@viagenie.ca
URI: http://viagenie.ca