

Network Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: April 18, 2013

G. Chen  
China Mobile  
October 15, 2012

Graceful IPv4 Sunset with Traffic Migration  
draft-chen-sunset4-traffic-migration-00

Abstract

In order to make a graceful IPv4 sunset, this memo described a method helping traffic migration to IPv6. With the growth of IPv6 traffic, operators could safely turn off IPv4 and evolve to IPv6-only network. In order to achieve the goal, new traffic-migration options have been proposed in DHCPv6 and PCP. IPv6 traffic steering could be performed using those configurations.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 18, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as

described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
2. Requirements Language . . . . .	3
3. Traffic Migration Technologies . . . . .	4
4. Configurations with DHCPv6 Options . . . . .	5
5. Configurations with PCP Options . . . . .	5
6. Security Considerations . . . . .	6
7. IANA Considerations . . . . .	6
8. References . . . . .	6
8.1. Normative References . . . . .	6
8.2. Informative References . . . . .	7
Author's Address . . . . .	7

## 1. Introduction

The working group of Sunset4 was targeted to standardize technologies that facilitate the graceful sunsetting of the IPv4 Internet in the context of the exhaustion of IPv4 address space while IPv6 is deployed. This memo has described the way to incrementally turn off the IPv4 by steering traffic to IPv6 networks.

As imminent demands to IP address, the community has to seek a way to accelerate IPv6. However, the tremendous success of the Internet has adhered to IPv4 technologies. ISPs don't want to significantly changed its IPv4 network. Dual stack[RFC4213] was designed to provide complete support for both Internet protocols. It's the simplest deployment model to enable IPv4 hosts to access the IPv4 Internet and IPv6 hosts to access the IPv6 Internet. With the thoughtful considerations, e.g. happy eyeballs[RFC6555], white-listing[RFC6589], dual-stack approach could ensure user experiences as original as possible.

[RFC6180]recommended the native dual-stack connectivity model. Some ISPs have already successfully deployed dual-stack networks, in which the dual-stack capable devices integrate both IPv6 and IPv4 forwarding. In those cases, IPv4 and IPv6 data flows are ships-in-the-night. [RFC6264]commentated such transition mechanism may be lack of drive to motive IPv6 growth, since most end users are not sufficiently expert to configure or maintain host-based IPv6 transition. If there are no IPv4 sunset technologies, IPv4 connectivity and traffic would still continue to represent the majority of traffic in most ISP networks.

The IPv4 sunset should be graceful. The arbitrary IPv4 turning off may don't help the IPv6 acceleration, but exacerbate the situation of instable IPv6 connections and IPv4 incompatibility. [RFC6586] has stated the concerns in a IPv6-only environment. It should be avoided during the period of IPv4 sunset, especially in a commercial network. Under those considerations, traffic migration could achieve the graceful process with no impacts to services. This memo enumerates several migration technologies in Section 3. The corresponding configurations have been described afterwards.

## 2. Requirements Language

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

### 3. Traffic Migration Technologies

With the stress of IP address shortage, switching the whole ISP network into IPv6-only would be considered a ultimate strategy. A number of IPv6 transition technologies were proposed. Some of them may likely be less optimal than equivalent technologies for native IP connections, i.e. IPv6-only and dual-stack networks. Whereas, it could help migrate IPv4 traffic to IPv6 network that is transparent to user's experiences. The Figure show the architecture those technologies apply to .

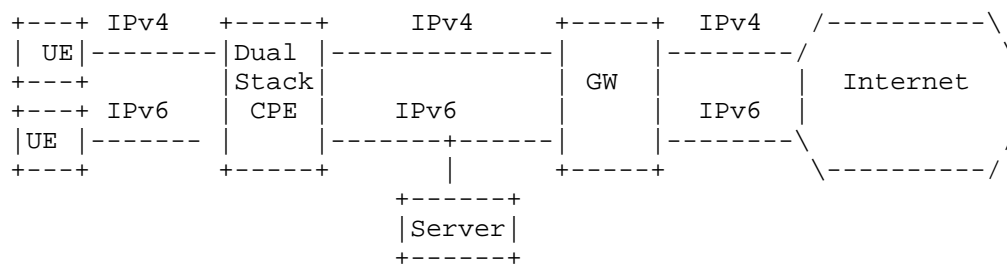


Figure 1: Traffic Migration architecture

Traffic migration technologies could shift IPv4 traffic to IPv6 links. Meanwhile, the issues of IPv4 compability have been thoroughly considered and addressed in those technologies. The migration enforcement could be located on a end-host or dual-stack CPE. Translations or tunnel could be performed at an enforcement point. Following enumerates relevant technologies.

- o Dual-stack Lite: it employs IPv4 over IPv6 tunnel on CPE. The packages would be encapsulated in IPv6 and transmitted. GW would decapsulate the IPv6 packages and perform IPv4/IPv4 NAT[RFC6333]. It should be noted that several technologies have been discussed in Softwires working group recently. Those technologies could also successfully switch traffic to IPv6 network.
- o 464xlat: it employs double translation framework[I-D.ietf-v6ops-464xlat]. CPE could receive IPv4 packages and make stateless translation[RFC6145] to IPv6. GW adopts stateful NAT64 [RFC6146]processing.
- o BIH: It employs host based translation[RFC6535]. Embedded BIH module could translate IPv4 packages into IPv6 on a host. Such process is transparent to IPv4 applications.

At a sunset stage, a devices(e.g. a host or CPE) would observe the appearance of enabling messages to discover the availability of

migration technology. Thus, when an ISP decides to switch their traffic to IPv6, the devices would detect and switch automatically to traffic-migration mode.

#### 4. Configurations with DHCPv6 Options

Enabling traffic migration could be achieved via DHCPv6. The migration DHCPv6 option is proposed as below to inform the device performing the traffic steering process. The format of the migration option is shown in Figure 2.

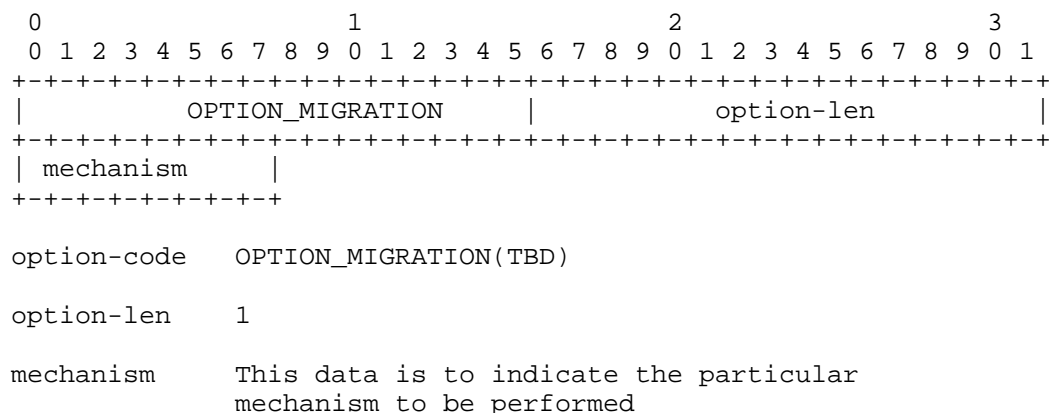


Figure2: Migration Option for DHCPv6

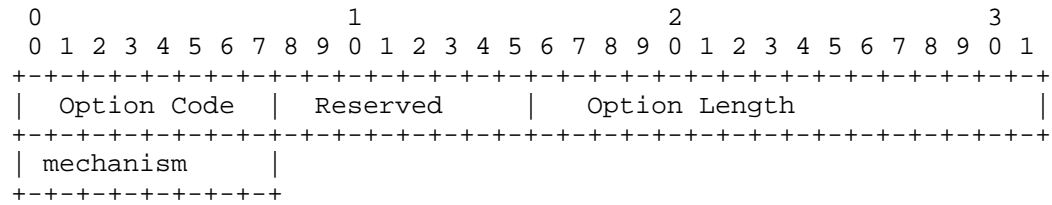
The DHCPv6 client MUST include the OPTION\_Migration option code in the Option Request Option[RFC3315].

[Editor note: the mechanism filed informs the device that the specific technology should be taken. This is very depending on the ISP strategy and implementations. Weighting different options is surely going beyond the scope of this document. Therefore, it should be decided whether the particular semantics should be defined in the draft.]

#### 5. Configurations with PCP Options

It's also feasible to deliver such message in a NAT environment, where there is coexistence of NAT44 and NAT64 on a network side. If PCP clients are embedded in CPE or UE, new PCP options could help to indicate migration preferring.

The format of migration PCP Option is depicted in Figure 3.



option-code    To be assigned by IANA

option-len     1

mechanism      This data is to indicate the particular  
                 mechanism to be performed

Figure3: Migration Option for PCP

A PCP Client MAY include a migration PCP Option in a MAP request to learn network capability used by an upstream PCP-controlled device. A PCP server controlling a NAT SHOULD be configured to return the value to indicate if the migration technology should be enable. When allowed, migration PCP Option conveys the value for the selection of specific mechanism.

[Editor note: Same concern applies to the mechanism filed. it should be decided whether the particular semantics should be defined in the draft. ]

## 6. Security Considerations

TBD

## 7. IANA Considerations

This document makes no request of IANA.

## 8. References

### 8.1. Normative References

[I-D.ietf-v6ops-464xlat]

Mawatari, M., Kawashima, M., and C. Byrne, "464XLAT: Combination of Stateful and Stateless Translation", draft-ietf-v6ops-464xlat-08 (work in progress), September 2012.

- [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.
- [RFC4213] Nordmark, E. and R. Gilligan, "Basic Transition Mechanisms for IPv6 Hosts and Routers", RFC 4213, October 2005.
- [RFC6145] Li, X., Bao, C., and F. Baker, "IP/ICMP Translation Algorithm", RFC 6145, April 2011.
- [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.
- [RFC6333] Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", RFC 6333, August 2011.
- [RFC6535] Huang, B., Deng, H., and T. Savolainen, "Dual-Stack Hosts Using "Bump-in-the-Host" (BIH)", RFC 6535, February 2012.

## 8.2. Informative References

- [RFC6180] Arkko, J. and F. Baker, "Guidelines for Using IPv6 Transition Mechanisms during IPv6 Deployment", RFC 6180, May 2011.
- [RFC6264] Jiang, S., Guo, D., and B. Carpenter, "An Incremental Carrier-Grade NAT (CGN) for IPv6 Transition", RFC 6264, June 2011.
- [RFC6555] Wing, D. and A. Yourtchenko, "Happy Eyeballs: Success with Dual-Stack Hosts", RFC 6555, April 2012.
- [RFC6586] Arkko, J. and A. Keranen, "Experiences from an IPv6-Only Network", RFC 6586, April 2012.
- [RFC6589] Livingood, J., "Considerations for Transitioning Content to IPv6", RFC 6589, April 2012.

Author's Address

Gang Chen  
China Mobile  
No.32 Xuanwumen West Street  
Xicheng District  
Beijing 100053  
China

Email: [phdgang@gmail.com](mailto:phdgang@gmail.com)





i»¿

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 sunseting IPv4, and identifies the gaps requiring additional work.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 28, 2018.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect

to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	2
2. Related Work . . . . .	3
3. Remotely Disabling IPv4 . . . . .	4
3.1. Indicating that IPv4 connectivity is unavailable . . . . .	4
3.2. Disabling IPv4 in the LAN . . . . .	4
4. Client Connection Establishment Behavior . . . . .	5
5. Disabling IPv4 in Operating System and Applications . . . . .	5
6. On-Demand Provisioning of IPv4 Addresses . . . . .	6
7. IPv4 Address Literals . . . . .	6
8. Managing Router Identifiers . . . . .	7
9. IANA Considerations . . . . .	7
10. Security Considerations . . . . .	7
11. Acknowledgements . . . . .	7
12. Informative References . . . . .	7
Annex A. Solution Ideas . . . . .	9
A.1. Remotely Disabling IPv4 . . . . .	9
A.1.1. Indicating that IPv4 connectivity is unavailable . . . . .	9
A.1.2. Disabling IPv4 in the LAN . . . . .	9
A.2. Client Connection Establishment Behavior . . . . .	10
A.3. Disabling IPv4 in Operating System and Applications . . . . .	10
A.4. On-Demand Provisioning of IPv4 Address. . . . .	10
A.5. Managing Router Identifiers . . . . .	10
Authors' Addresses . . . . .	11

## 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 LLNMR [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 choosing 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 ratio. 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: January 10, 2013

S. Perreault  
Viagenie  
T. Tsou  
Huawei Technologies (USA)  
S. Sivakumar  
Cisco Systems  
July 9, 2012

Managed Objects for Carrier Grade NAT (CGN)  
draft-perreault-sunset4-cgn-mib-00

Abstract

This memo defines a portion of the Management Information Base (MIB) that may be used for monitoring of a device capable of Carrier Grade NAT function.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 10, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as



described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
2. Terminology . . . . .	3
3. Overview . . . . .	3
4. Definitions . . . . .	3
5. Security Considerations . . . . .	9
6. IANA Considerations . . . . .	9
7. Normative References . . . . .	9
Authors' Addresses . . . . .	9

## 1. Introduction

[I-D.ietf-behave-nat-mib] defines objects for managing network address translators (NATs). This document builds on top of it, defining objects specifically for Carrier Grade NATs (CGN).

## 2. Terminology

The "CGN" term is defined in [I-D.ietf-behave-lsn-requirements].

## 3. Overview

New features in this module are as follows:

Per-subscriber counters, limits, and notifications: Carrier-Grade NATs operate with a notion of "subscriber", to which are associated a set of counters, limits, and notifications. The subscriber identifier may not necessarily be an internal address, as in the case of DS-Lite, where the identifier is the IPv6 address of the tunnel endpoint and the internal addresses are the same for each subscriber.

## 4. Definitions

The following objects are added to the MIB module defined in [I-D.ietf-behave-nat-mib].

-- notifications

```
newNatNotifSubscriberMappings NOTIFICATION-TYPE
  OBJECTS { newNatSubscriberCntMappings }
  STATUS current
  DESCRIPTION
    "This notification is generated when newNatSubscriberCntMappings
     exceeds the value of newNatSubscriberMapNotifyThresh, unless
     newNatSubscriberMapNotifyThresh is zero.."
  ::= { newNatNotifications 5 }
```

-- limits

```
newNatLimitSubscribers OBJECT-TYPE
  SYNTAX Unsigned32
  MAX-ACCESS read-write
  STATUS current
```

```
DESCRIPTION
    "Global limit on the number of subscribers with active mappings.
    Zero means unlimited."
 ::= { newNatLimits 6 }

-- subscribers

newNatSubscribers OBJECT IDENTIFIER ::= { newNatObjects 5 }

newNatSubscribersTable OBJECT-TYPE
    SYNTAX SEQUENCE OF NewNatSubscribersTableEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Table of CGN subscribers."
    ::= { newNatSubscribers 1 }

newNatSubscribersTableEntry OBJECT-TYPE
    SYNTAX NewNatSubscribersTableEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Each entry describes a single CGN subscriber."
    INDEX { newNatSubscriberIdentifierType,
            newNatSubscriberIdentifier }
    ::= { newNatSubscribersTable 1 }

NewNatSubscribersTableEntry ::=
    SEQUENCE {
        newNatSubscriberIdentifierType    InetAddressType,
        newNatSubscriberIdentifier        InetAddress,
        newNatSubscriberIntPrefixType     InetAddressType,
        newNatSubscriberIntPrefix         InetAddress,
        newNatSubscriberIntPrefixLength   InetAddressPrefixLength,
        newNatSubscriberPool              NatPoolIndex,
        newNatSubscriberCntTranslates     Counter64,
        newNatSubscriberCntOOP            Counter64,
        newNatSubscriberCntResource       Counter64,
        newNatSubscriberCntStateMismatch Counter64,
        newNatSubscriberCntQuota          Counter64,
        newNatSubscriberCntMappings       Gauge32,
        newNatSubscriberCntMapCreations   Counter64,
        newNatSubscriberCntMapRemovals    Counter64,
        newNatSubscriberLimitMappings     Unsigned32,
        newNatSubscriberMapNotifyThresh   Unsigned32
    }
```

```
newNatSubscriberIdentifierType OBJECT-TYPE
    SYNTAX InetAddressType
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Address type of the subscriber identifier."
    ::= { newNatSubscribersTableEntry 1 }

newNatSubscriberIdentifier OBJECT-TYPE
    SYNTAX InetAddress (SIZE (4|16))
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Address used for uniquely identifying the subscriber.

        In traditional NAT, this is the internal address assigned to
        the CPE. In case an address range is assigned to a subscriber,
        the first address in the range is used as identifier. For
        tunnelled connectivity (e.g., DS-Lite [RFC6333]), the outer
        address is used as identifier (i.e., the IPv6 address in the
        case of DS-Lite)."
    ::= { newNatSubscribersTableEntry 2 }

newNatSubscriberIntPrefixType OBJECT-TYPE
    SYNTAX InetAddressType
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Subscriber's internal prefix type."
    ::= { newNatSubscribersTableEntry 3 }

newNatSubscriberIntPrefix OBJECT-TYPE
    SYNTAX InetAddress
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Prefix assigned to a subscriber's CPE."
    ::= { newNatSubscribersTableEntry 4 }

newNatSubscriberIntPrefixLength OBJECT-TYPE
    SYNTAX InetAddressPrefixLength
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Length of the prefix assigned to a subscriber's CPE, in bits.
        In case a single address is assigned, this will be 32 for IPv4
        and 128 for IPv6."
    ::= { newNatSubscribersTableEntry 5 }
```

## newNatSubscriberPool OBJECT-TYPE

SYNTAX NatPoolIndex

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"External address pool to which this subscriber belongs."

::= { newNatSubscribersTableEntry 6 }

## newNatSubscriberCntTranslates OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of packets received from or sent to this subscriber and to which NAT has been applied."

::= { newNatSubscribersTableEntry 7 }

## newNatSubscriberCntOOP OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of packets received from this subscriber to which NAT could not be applied because no external port was available, excluding quota limitations."

::= { newNatSubscribersTableEntry 8 }

## newNatSubscriberCntResource OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of packets received from this subscriber to which NAT could not be applied because of resource constraints (excluding out-of-ports condition)."

::= { newNatSubscribersTableEntry 9 }

## newNatSubscriberCntStateMismatch OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of packets received from or destined to this subscriber to which NAT could not be applied because of mapping state mismatch. For example, a TCP packet that matches an existing mapping but is dropped because its flags are incompatible with the current state of the mapping would cause this counter to be incremented."

```
 ::= { newNatSubscribersTableEntry 10 }

newNatSubscriberCntQuota OBJECT-TYPE
    SYNTAX Counter64
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "The number of packets received from or destined to this
        subscriber to which NAT could not be applied because of quota
        limitations. Quotas include absolute limits as well as limits
        on the rate of allocation."
    ::= { newNatSubscribersTableEntry 11 }

newNatSubscriberCntMappings OBJECT-TYPE
    SYNTAX Gauge32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Number of currently active mappings created by or for this
        subscriber.

        Equal to newNatSubscriberCntMapRemovals -
        newNatSubscriberCntMapCreations."
    ::= { newNatSubscribersTableEntry 12 }

newNatSubscriberCntMapCreations OBJECT-TYPE
    SYNTAX Counter64
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Number of mappings created by or for this subscriber."
    ::= { newNatSubscribersTableEntry 13 }

newNatSubscriberCntMapRemovals OBJECT-TYPE
    SYNTAX Counter64
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Number of mappings removed by or for this subscriber."
    ::= { newNatSubscribersTableEntry 14 }

newNatSubscriberLimitMappings OBJECT-TYPE
    SYNTAX Unsigned32
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "Limit on the number of active mappings created by or for this
        subscriber. Zero means unlimited."
```

```
 ::= { newNatSubscribersTableEntry 15 }

newNatSubscriberMapNotifyThresh OBJECT-TYPE
    SYNTAX Unsigned32
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "See newNatNotifSubscriberMappings."
    ::= { newNatSubscribersTableEntry 16 }

-- conformance groups

newNatGroupSubscriberObjects OBJECT-GROUP
    OBJECTS { newNatSubscriberIntPrefixType,
               newNatSubscriberIntPrefix,
               newNatSubscriberIntPrefixLength,
               newNatSubscriberPool,
               newNatSubscriberCntTranslates,
               newNatSubscriberCntOOP,
               newNatSubscriberCntResource,
               newNatSubscriberCntStateMismatch,
               newNatSubscriberCntQuota,
               newNatSubscriberCntMappings,
               newNatSubscriberCntMapCreations,
               newNatSubscriberCntMapRemovals,
               newNatSubscriberLimitMappings,
               newNatSubscriberMapNotifyThresh,
               newNatLimitSubscribers }
    STATUS current
    DESCRIPTION
        "Per-subscriber counters, limits, and thresholds."
    ::= { newNatGroups 4 }

-- compliance statements

newNatCGNCompliance MODULE-COMPLIANCE
    STATUS current
    DESCRIPTION
        "NATs that have 'Paired IP address pooling' and 'Receive
        Fragments Out of Order' behavior [RFC4787] and implement the
        objects in this group can claim this level of compliance.

        This level of compliance is to be expected of a CGN compliant
        with [I-D.ietf-behave-lsn-requirements]."
```

```
MODULE -- this module
    MANDATORY-GROUPS { newNatGroupBasicObjects,
```

```
newNatGroupBasicNotifications,  
newNatGroupAddrMapObjects,  
newNatGroupAddrMapNotifications,  
newNatGroupFragmentObjects,  
newNatGroupSubscriberObjects,  
newNatGroupSubscriberNotifs }  
 ::= { newNatCompliance 4 }
```

## 5. Security Considerations

TBD

## 6. IANA Considerations

TBD

## 7. Normative References

[I-D.ietf-behave-lsn-requirements]  
Perreault, S., Yamagata, I., Miyakawa, S., Nakagawa, A.,  
and H. Ashida, "Common requirements for Carrier Grade NATs  
(CGNs)", draft-ietf-behave-lsn-requirements-07 (work in  
progress), June 2012.

[I-D.ietf-behave-nat-mib]  
Perreault, S., Tsou, T., and S. Sivakumar, "Additional  
Managed Objects for Network Address Translators (NAT)",  
draft-ietf-behave-nat-mib-01 (work in progress),  
June 2012.

## Authors' Addresses

Simon Perreault  
Viagenie  
246 Aberdeen  
Quebec, QC G1R 2E1  
Canada

Phone: +1 418 656 9254  
Email: [simon.perreault@viagenie.ca](mailto:simon.perreault@viagenie.ca)  
URI: <http://viagenie.ca>



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

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

Senthil Sivakumar  
Cisco Systems  
7100-8 Kit Creek Road  
Research Triangle Park, North Carolina 27709  
USA

Phone: +1 919 392 5158  
Email: ssenthil@cisco.com



Network Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: January 16, 2014

S. Perreault  
Viagenie  
W. George  
Time Warner Cable  
T. Tsou  
Huawei Technologies (USA)  
T. Yang  
L. Li  
China Mobile  
July 15, 2013

Turning off IPv4 Using DHCPv6 or Router Advertisements  
draft-perreault-sunset4-noipv4-03

Abstract

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

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 16, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect

to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	2
2. Terminology . . . . .	3
3. The Problems We're Trying to Fix . . . . .	4
3.1. Load on DHCPv4 Server . . . . .	4
3.2. Bandwidth Consumption . . . . .	4
3.3. Power Inefficiency . . . . .	4
3.4. IPv4 only Applications . . . . .	4
4. Design Considerations . . . . .	4
4.1. DHCPv6 vs DHCPv4 . . . . .	4
4.2. DHCPv6 vs RA . . . . .	5
5. The No-IPv4 Option . . . . .	6
5.1. DHCPv6 Wire Format . . . . .	6
5.2. RA Wire Format . . . . .	6
5.3. Semantics . . . . .	7
5.4. Example . . . . .	9
6. Security Considerations . . . . .	10
7. IANA Considerations . . . . .	10
8. Acknowledgements . . . . .	10
9. References . . . . .	10
9.1. Normative References . . . . .	10
9.2. Informative References . . . . .	11
Appendix A. Test Results of Terminals Behavior . . . . .	11
Authors' Addresses . . . . .	12

## 1. Introduction

When a dual-stack host makes a DHCPv4 request, it typically interprets the absence of a response as a failure condition. This makes it difficult to deploy such nodes in an IPv6-only network.

Take for example a home router that is dual-stack capable but provisioned with an IPv6-only WAN connection. When the router boots, it typically assigns an IPv4 address to its LAN interface, starts services on that interface, and starts handing 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, in turn, install a default route pointing to the router and start behaving 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 when it is desired that it be turned off.

The other example is about DHCPv4 server. In Dual-Stack LAN/WLAN network or intranet, the core router or AC often plays the role of DHCP server, and the clients are server thousands PC or mobile phones. If the server is configured in IPv6-only, the dual-stack or IPv4-only clients will broadcast DHCPDISCOVER messages endlessly in the LAN or WLAN. The thousands clients will cause a DDOS-like attack to all the servers in the network. Since there are not specific descriptions in any RFCs for client's behavior when it does not receive the DHCPOFFER in response to its DHCPDISCOVER message, various OS deploy different backoff algorithms. We tested server popular OS(es), the test results is listed in the appendix.

A new mechanism is needed to indicate the absence of IPv4 connectivity or service that the goal is turning off IPv4, this new signaling mechanism shall be transported over IPv6. Therefore, we 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. The Problems We're Trying to Fix

#### 3.1. Load on DHCPv4 Server

When a DHCPv4 server is present but intentionally does not respond to a dual-stack node, the aggregated traffic generated by multiple such dual-stack nodes can represent a significant useless load. This scenario can be encountered for example with an ISP serving multiple types of subscribers where some will get IPv4 addresses and others not. It might not be feasible for operational reasons to block the useless requests before they reach the DHCPv4 server, e.g. if the DHCPv4 server itself is the one that has knowledge about which node should or should not get an IPv4 address.

#### 3.2. Bandwidth Consumption

In addition to useless load on the DHCPv4 server, the above scenario could also consume a significant amount of bandwidth, particularly if the aggregated traffic from many clients goes through a low-bandwidth 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 IPv4 LLA to bombard the LAN with IPv4 packets. In an IPv6-only environment, it can get quite annoying and waste a lot 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 for turning off IPv4. An equivalent option could conceivably be created for DHCPv4. Here is a discussion of the pros and cons. 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 it is not as clear for a DHCPv4 option. 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 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 exclusively 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.

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.

## 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		Windows XP		IOS_5.0.1		Android_2.3.7		Symbian_S60	
	Time	Time offset	Time	Time offset	Time	Time offset	Time	Time offset	Time	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  
Viagenie  
246 Aberdeen  
Quebec, QC G1R 2E1  
Canada

Phone: +1 418 656 9254  
Email: [simon.perreault@viagenie.ca](mailto:simon.perreault@viagenie.ca)  
URI: <http://viagenie.ca>

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

Email: [wesley.george@twcable.com](mailto: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](mailto:tina.tsou.zouting@huawei.com)

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

Email: [yangtianle@chinamobile.com](mailto:yangtianle@chinamobile.com)

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

Email: [lilianyuan@chinamobile.com](mailto:lilianyuan@chinamobile.com)

Network Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: April 25, 2013

T. Tsou  
Huawei Technologies (USA)  
W. Liu  
Huawei Technologies  
S. Perreault  
Viagenie  
R. Penno  
Cisco Systems, Inc.  
M. Chen  
FreeBit  
October 22, 2012

Stateless IPv4 Network Address Translation  
draft-tsou-stateless-nat44-02

Abstract

This memo describes a protocol for decentralizing IPv4 NAT to the customer-premises equipment (CPE) such that no state information is kept on the central NAT device. The CPE uses a restricted source port set that is encoded in its provisioned IPv4 WAN address. The NAT device performs only strictly stateless address (not port) translation.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 25, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal



Provisions Relating to IETF Documents  
(<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
2. Terminology . . . . .	3
3. Address Formats . . . . .	4
4. Customer Provisioning . . . . .	5
5. SLNAT44 Configuration . . . . .	6
6. Port Set Computation . . . . .	6
7. CPE Operation . . . . .	7
7.1. ALG Handling . . . . .	8
8. SLNAT44 Operation . . . . .	8
8.1. Internal to External . . . . .	8
8.2. External to Internal . . . . .	8
8.3. Fragment Handling . . . . .	9
9. Address Mapping Example . . . . .	9
10. Security Considerations . . . . .	10
11. Acknowledgements . . . . .	10
12. References . . . . .	10
12.1. Normative References . . . . .	10
12.2. Informative References . . . . .	10
Authors' Addresses . . . . .	11

## 1. Introduction

IPv4 address exhaustion has become world-wide reality. NAT is one of the solutions to deal with the problem. The drawbacks of traditional NAT include statefulness and the need to track transport-layer sessions. This makes NAT complex, hard to scale up, and fragile.

This document describes a method of deploying stateless NAT as a backwards-compatible evolution of an IPv4-only network.

The assumed topology is illustrated in Figure 1.

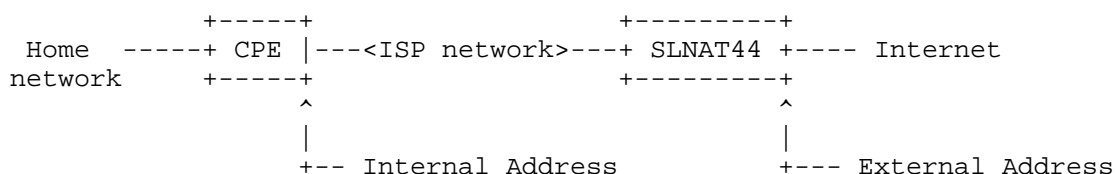


Figure 1: Stateless NAT44 topology

When CPE is configured working as a transparent bridge, internal addresses are directly assigned to the end hosts in the home network, as is shown in Figure 2.

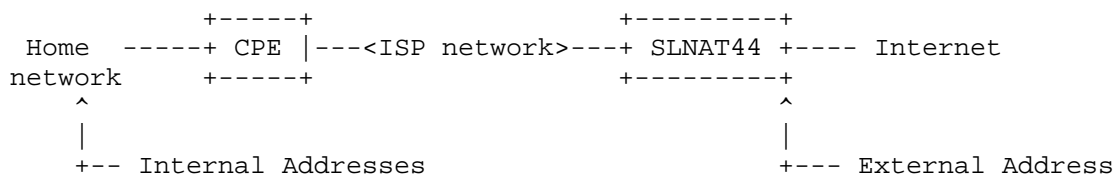


Figure 2: Stateless NAT44 topology: CPE as bridge

Note that SLNAT44 has no IPv6 component. Any deployment of IPv6 is unaffected by SLNAT44. Therefore, this document only describes IPv4 addresses and IPv4 packets. IPv6 is not discussed further.

## 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 used throughout this document:

Port set: Set of transport-layer ports that each CPE is assigned, to be used as source ports by packets sent by the CPE.

Port Set ID: A value from which a unique port set is algorithmically derived.

SLNAT44: Depending on the context, either the stateless NAT44 protocol or the stateless NAT44 device that translates between internal and external addresses. NAT44 in turn stands for "IPv4-to-IPv4 NAT".

Internal Address: The IPv4 address assigned to a CPE. It is used in the ISP network between the CPE and the SLNAT44.

External Address: The IPv4 address used on the Internet and routed to the SLNAT44.

Mapping rule: A set of parameters configured on the SLNAT44 (not on the CPE) describing the relationship between internal and external addresses.

### 3. Address Formats

Internal addresses have the format illustrated in Figure 3. The addresses are simply made of three parts concatenated together: the Internal Prefix, the External Suffix, and the Port Set ID.

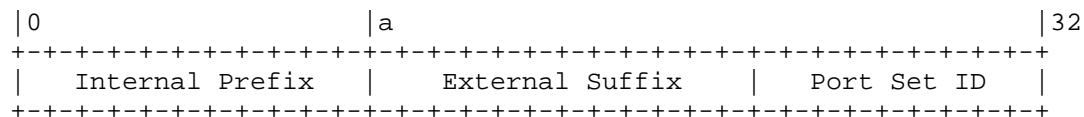


Figure 3: Internal Address format

External Addresses have the format illustrated in Figure 4. It is made of two parts: the External Prefix and the External Suffix.

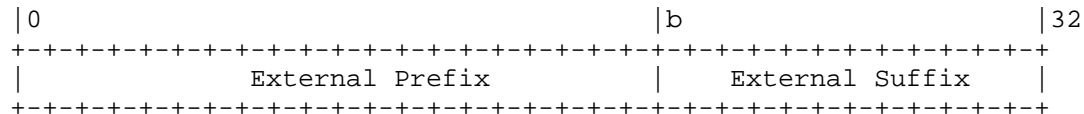


Figure 4: External Address format

The lengths of the Internal and External Prefixes, "a" and "b", are mandatory parameters of SLNAT44. They are determined by the ISP. They need not be communicated to the CPE. Other lengths can be

computed from them as follows:

- o Length of External Suffix:  $32 - b$
- o Length of Port Set ID:  $b - a$

#### 4. Customer Provisioning

Customer Provisioning is applied to the CPE when the CPE serves a gateway.

As part of its start up routine, the CPE is assigned an IPv4 address by the ISP using regular means (DHCP, PPP, etc.). This is the Internal Address.

In addition, using new provisioning options, the CPE is assigned a Port Set ID.

Optionally, a Port Set Mask is also provisioned to the CPE. This mask is of the same length as the Port Set ID (i.e.,  $b-a$  bits). Its purpose is to allow discontinuous port ranges. If no mask is provided, a mask of all ones is assumed by default, which implies a continuous port range. System ports (0-1023) should not to be assigned to any CPE.

In summary, the CPE is provisioned with the following elements:

- o IPv4 address (as usual)
- o Port Set ID
- o Port Set Mask (optional)

When the CPE is configured in the bridge mode, all the above features are provisioned directly to the end host behind the CPE.

Note: no matter in which mode the CPE is running, the customer provisioning could be either dynamic or static. Static provisioning implies an address planning for the private IPv4 address (i.e., RFC1918 addresses) inside in the domain. Static provisioning enables servers (passive daemons) at the home network being accessible within the domain. CPE running as bridge makes this feature easy to deploy while running as L3 gateway requires port redirection if an in-domain server at a host is demanded.

## 5. SLNAT44 Configuration

The SLNAT44 is configured with a set of mapping rules. Each rule contains:

- o Internal Prefix
- o External Prefix
- o Port Set Mask (optional)

Prefixes include their length. For simplicity, rule prefixes MUST NOT overlap with other rules.

If it is absent, the Port Set Mask is assumed to be all ones by default.

## 6. Port Set Computation

Given a Port Set ID and a Port Set Mask, both n bits in length, the set of allowed ports is defined as the set of port numbers for which the higher-order n bits of their binary expression whose corresponding mask bits are 1 are equal to corresponding bits from the Port Set ID.

```

|0          |5
+---+---+---+
|1 1 1 0 1|  Port Set ID = 29 (length n = 5 bits)
+---+---+---+
& & & &
+---+---+---+
|1 1 1 1 1|  Port Set Mask
+---+---+---+
| | | | |
V V V V V
+---+---+---+---+---+---+---+---+---+---+---+---+
|1 1 1 0 1 x x x x x x x x x x x x|  Port Set = 59392-61439
+---+---+---+---+---+---+---+---+---+---+---+---+
|0          |16

```

Figure 5: Example Contiguous Port Set Computation

```

|0                                     |8
+---+---+---+---+---+---+
|0 0 1 0 1 1 1 1| Port Set ID = 29 (length n = 8 bits)
+---+---+---+---+---+---+
& & & & & & &
+---+---+---+---+---+---+
|0 0 1 1 1 1 1 1| Port Set Mask
+---+---+---+---+---+---+
| | | | | | | |
V V V V V V V V
+---+---+---+---+---+---+
|x x 1 0 1 1 1 1 x x x x x x x x| Port Set = 12032-12287, 28416-28671,
+---+---+---+---+---+---+      44800-45055, 61184-61439
|0                                     |16

```

Figure 6: Example Non-Contiguous Port Set Computation

It follows that the number of ports in the set is  $2^{(16-x)}$ , where  $x$  is the number of ones in the Port Set Mask.

This computation is performed by the CPE as part of its provisioning routine as well as by the SLNAT44 for dropping packets with ports outside the allowed range.

For the purposes of SLNAT44, a "source port" corresponds to either a TCP source port, a UDP source port, or an ICMPv4 identifier, while a "destination port" corresponds to either a TCP destination port, a UDP destination port, or an ICMPv4 identifier. Note that an ICMPv4 identifier plays the role of both source and destination port.

Transport protocols other than TCP and UDP, as well as ICMPv4 types without an identifier field, are not supported.

## 7. CPE Operation

CPE can be configured as either a gateway or transparent bridge.

In the gateway mode, packets sent from the CPE MUST have the provisioned IPv4 address as source and MUST have a source port that is within the allowed set. This is usually accomplished by having the CPE run a NAT44 configured with the provisioned address and allowed port set and having it process all packets sent out the WAN interface.

Packets received by the CPE on its WAN interface with a destination port outside the allowed range MUST be dropped.

In the bridge mode, however, CPE only transfers packets and therefore the service of stateless NAT44 is performed by the SLNAT44 directly towards end hosts that possibly running as in-domain servers.

Regardless of any mode of the CPE, the operation involves injecting private addresses (or prefixes) into the intra-domain backbone routing infrastructure. It is necessary to operationally ensure that there are no private addresses/prefixes are leaking into the backbone route tables unless they are assigned by the SLNAT44 to CPEs or directly to hosts.

### 7.1. ALG Handling

If the CPE implements application level gateways (ALGs) such as FTP, RSTP or PPTP, it must ensure that ports present in the payload when translated fall within the allowed range.

## 8. SLNAT44 Operation

### 8.1. Internal to External

When it receives a packet on an internal interface, the SLNAT44 finds the rule whose Internal Prefix matches the packet's source address. It extracts the Port Set ID from the packet's source address. It then checks if the packet's source port is within the allowed set, using the rule's Port Set Mask. If it is not, the packet MUST be dropped.

If the packet's source port is within the allowed set, the SLNAT44 builds the External Address by concatenating the rule's External Prefix with the External Suffix extracted from the packet's source address. It then replaces the packet's source address with this External Address. The IPv4 and transport-layer checksums are updated as necessary. The packet is then forwarded as usual.

### 8.2. External to Internal

When it receives a packet on an external interface, the SLNAT44 finds the rule whose External Prefix matches the packet's destination address. It then builds the Internal Address by concatenating the rule's Internal Prefix, the External Suffix extracted from the packet's destination address, and the Port Set ID computed by applying the rule's Port Set Mask to the packet's destination port's higher-order bits. It then replaces the packet's destination address with this Internal Address. The IPv4 and transport-layer checksums are updated as necessary. The packet is then forwarded as usual.

### 8.3. Fragment Handling

If the incoming IP packet contains a fragment, then more processing may be needed. This specification leaves open the exact details of how a SLNAT44 handles incoming IP packets containing fragments, and simply requires that the external behavior of the SLNAT44 be compliant with the following conditions.

The SLNAT44 **MUST** handle fragments. In particular, SLNAT44 **MUST** handle fragments arriving out of order, conditional on the following:

- o The SLNAT44 **MUST** limit the amount of resources devoted to the storage of fragmented packets in order to protect from DoS attacks.
- o As long as the SLNAT44 has available resources, the SLNAT44 **MUST** allow the fragments to arrive over a time interval. The time interval **SHOULD** be configurable and the default value **MUST** be of at least 2 seconds.
- o The SLNAT44 **MAY** require that the UDP, TCP, or ICMPv4 header be completely contained within the fragment that contains fragment offset equal to zero.

For incoming packets carrying TCP or UDP fragments with a non-zero checksum, SLNAT44 **MAY** elect to queue the fragments as they arrive and translate all fragments at the same time. In this case, the incoming tuple is determined as documented above to the un-fragmented packets. Alternatively, a SLNAT44 **MAY** translate the fragments as they arrive, by storing information that allows it to compute the necessary port number for fragments other than the first. In the latter case, subsequent fragments may arrive before the first, and the rules (in the bulleted list above) about how the SLNAT44 handles (out-of-order) fragments apply.

Implementers of SLNAT44 should be aware that there are a number of well-known attacks against IP fragmentation; see [RFC1858] and [RFC3128]. Implementers should also be aware of additional issues with reassembling packets at high rates, described in [RFC4963].

### 9. Address Mapping Example

An operator has two public ranges of size /18 and /19 called foo and bar respectively. It plans to use 10/8 as its internal address prefix and PSID (port range) of length 5. Two prefixes of the internal network



The internal prefixes lengths are:

- o 32 - 18 - 5 = 13 (derived from foo)
- o 32 - 19 - 5 = 14 (derived from bar)

This will give the following possible mappings:

- o foo/18 <--> 10.0.0.0/13
- o bar/19 <--> 10.128.0.0/14

Author note: Discuss the where internal prefixes are overlapping

## 10. Security Considerations

The security considerations related to IP address sharing documented in RFC 6269 [RFC6269] and RFC 6056 [RFC6056] apply to SLNAT44.

## 11. Acknowledgements

Section 8.3 is adapted from [RFC6146].

## 12. References

### 12.1. Normative References

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

### 12.2. Informative References

- [RFC1858] Ziemba, G., Reed, D., and P. Traina, "Security Considerations for IP Fragment Filtering", RFC 1858, October 1995.
- [RFC3128] Miller, I., "Protection Against a Variant of the Tiny Fragment Attack (RFC 1858)", RFC 3128, June 2001.
- [RFC4963] Heffner, J., Mathis, M., and B. Chandler, "IPv4 Reassembly Errors at High Data Rates", RFC 4963, July 2007.
- [RFC6056] Larsen, M. and F. Gont, "Recommendations for Transport-Protocol Port Randomization", BCP 156, RFC 6056, January 2011.

- [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.
- [RFC6269] Ford, M., Boucadair, M., Durand, A., Levis, P., and P. Roberts, "Issues with IP Address Sharing", RFC 6269, June 2011.

## Authors' Addresses

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

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

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

Email: liushucheng@huawei.com

Simon Perreault  
Viagenie  
246 Aberdeen  
Quebec, QC G1R 2E1  
Canada

Email: simon.perreault@viagenie.ca

Reinaldo Penno  
Cisco Systems, Inc.  
170 West Tasman Drive  
San Jose, California 95134  
USA

Email: repenno@cisco.com

Maoke Chen  
FreeBit  
3-6 Maruyama-cho  
Shibuya-ku, Tokyo 150-0044  
Japan

Email: fibrib@gmail.com



Internet Engineering Task Force  
Internet-Draft  
Intended status: Standards Track  
Expires: April 15, 2013

T. Yang  
L. Li  
Q. Ma  
China Mobile  
Oct 12, 2012

Weakening Aggregated Traffic of DHCP Discover Messages  
draft-yang-sunset4-weaken-dhcp-00

Abstract

This document proposes two methods to mitigate aggregated traffic caused by discover messages the dual stack host send to the server.

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 15, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
2. Requirements Language . . . . .	3
3. Potential Problems . . . . .	3
4. DHCPv6 solution . . . . .	6
5. RA solution . . . . .	7
6. Security Considerations . . . . .	8
7. IANA Considerations . . . . .	8
8. References . . . . .	8
Authors' Addresses . . . . .	8

## 1. Introduction

In RFC3315 [RFC3315, DHCPv6], SOL\_MAX\_RT is defined in DHCPv6 to prevent the frequently requesting of clients, which reduce 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 some cases, this will lead to an unacceptably high volum of aggregated traffic at a DHCP server, especially in the "Dual-Stack host/network + IPv6-Only DHCP server" scenario:

As everyone knows, our network is changing from IPv4-Only to Dual-Stack, and even IPv6-Only in the near future. We may turn off some IPv4 services gradually, such as DHCP. If a Dual-Stack host initials DHCP Discover messages through the link to a DHCPv6-Only server, it cannot get any response. Then the host will re-broadcast the messages endlessly, that may cause the aggregated traffic.

In this document, we propped two methods to solve this problem, creating a new option in DHCPv6 or in RS/RA, described as below.

## 2. Requirements Language

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

## 3. Potential Problems

RFC2131 [RFC2131] defines the interaction between the DHCP server and clients. There are no specific discription for client's operation when the client does not receive the DHCPOFFER in response to its DHCPDISCOVER message. In normal IPv4 environment, clients will flood DHCPDISCOVER messages only when the server or link is broken. But in Dual-Stack scenarios, the problem becomes more frequent and serious.

In Dual-Stack LAN/WLAN network or intranet, the core router or AC often plays the role of DHCP server, and the clients are serval thousands PC or mobile phones. If the server is configured in IPv6-only, the dual-stack or IPv4-only clients will broadcast DHCPDISCOVER messages endlessly in the LAN or WLAN. The thousands clients will cause a DDOS-like attack to all the servers in the network.

This situation may occur when the networks or serveices gradually updated to IPv6-Only.

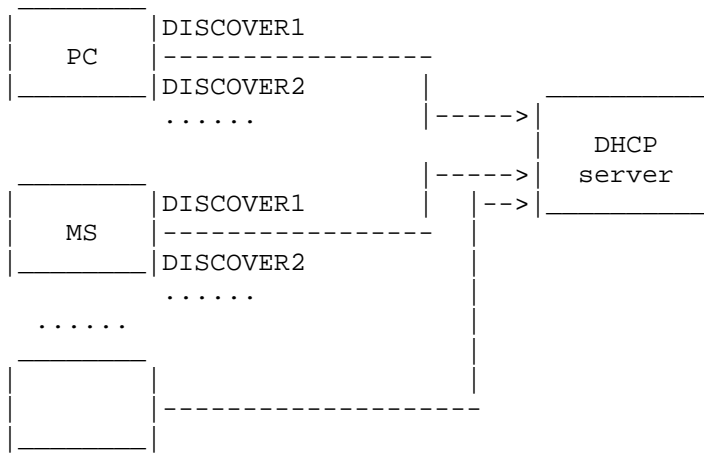


Figure 1: DHCPDISCOVER flood in LAN/WLAN

To avoid this problem, most of the terminals creat backoff algorithms which can help them retransmit DHCPDISCOVER message in different frequency according to their state machine in different Operating Systems, because there is no specific defenition in RFCs to restrict the terminals behaviors when the server is down or in a dual-stack scenario as discripted upwards. But the same point of almost all the verious Operating Systems is that they could not stop DHCPDISCOVER requests enven to an IPv6-only server. We test some of the most popular terminals' OS in WLAN, the results are illuminated as below.



DHCP Discovery Packages Time Table										
No	Windows7		Windows XP		IOS_5.0.1		Android_2.3.7		Symbian_S60	
	Time	Time offset	Time	Time offset	Time	Time offset	Time	Time offset	Time	Time offse
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

Figure2:Terminals DHCPDISCOVER requests when Server's  
DHCP module

is down

In figure 2:

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.

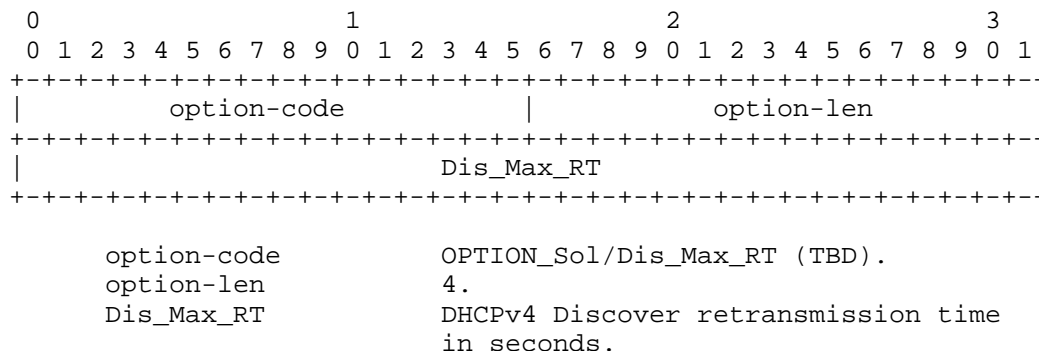
Android2.3.7 is the only Operating System which can stop DISCOVER

request by disconnect its wireless connection. It reboot wireless and dhcp connection every 20 seconds.

Obviously, DHCP server needs to weaken the traffic which is like DDoS attack caused by the clients when many DHCPv4 clients send discovery messages incessantly when the DHCPv4 server is configured no respond to Discover messages.

#### 4. DHCPv6 solution

According to the definition of DHCPv6 option in RFC3315 [RFC3315], a new option named OPTION\_Dis\_Max\_RT is defined to affect the retransmission of DHCPv4 DISCOVER message of the host. The format of OPTION\_Dis\_Max\_RT is:



#### OPTION\_Dis\_Max\_RT

The OPTION\_Dis\_Max\_RT option needs IANA to assign a new Code to indicate. Its length (Len value) is 4 octets. Dis\_Max\_RT is the value of DHCPv4 Discover message retransmission time in the unit of second.

If Dis\_Max\_RT=0, server will respond Offer or other DHCP messages in normal;

If Dis\_Max\_RT>0, server won't respond to Discover immediately, cliet should wait for resending Discover message later;

If Dis\_Max\_RT=FFFF, cliet should not send Discover message any more.

A DHCPv6 client MUST include the OPTION\_Dis\_Max\_RT code in Option Request Option [RFC3115, section 22.7]. The DHCPv6 server MAY

include the OPTION\_Dis\_Max\_RT in any response it sends to a client.

The process of this option is described below:

1. Client must initial the request code in the Option Request Option in the Discover messages.
2. When server receives a request, it MUST assign an appropriate value in the response to the client. It can set FFFF in the Dis\_Max\_RT field when the dhcp module is turned off or according to the administrator's configuration.

5. RA solution

Neighbor Discovery for IPv6 defined in RFC4861[RFC4861] is a basic protocol of IPv6. It is used more widely than DHCPv6. When the value of M in Router Advertisement(RA) message is set, DHCPv6 can only be set to active. If M and O are not set, RA will be used to deliver the IPv6 prefix instead of DHCPv6. A new option is defined in Router Advertisement(RA) messages to be used to avoid frequent retransmission.

According to the definition of RA option in RFC4861 [RFC4861], a new option named Option\_Dis\_Max\_RT is defined to affect the retransmission of DHCPv4 DISCOVER message.

The format of OPTION\_Dis\_Max\_RT 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\_Dis\_Max\_RT

The OPTION\_Dis\_Max\_RT option needs IANA to assign a new Code to

indicate and its length (Len value) is 8 octets. Dis\_Max\_RT is the value of DHCPv4 Discover message retransmission time in the unit of second.

If Dis\_Max\_RT=0, server will respond Offer or other DHCP messages in normal;

If Dis\_Max\_RT>0, server won't respond to Discover immediately, client should wait for resending Discover message later;

If Dis\_Max\_RT=FFFF, client should not send Discover message any more.

The process is a little simpler than DHCPv6:

1. Server send RA with this option to client to tell it the intervals to resend Discover messages.

## 6. Security Considerations

The security problem is under discussion.

## 7. IANA Considerations

IANA is requested to assign an option code from the "DHCP Option Codes" Registry for OPTION\_DIS\_MAX\_RT.

## 8. References

- (1) RFC[2131] Dynamic Host Configuration Protocol
- (2) RFC[3315] Dynamic Host Configuration Protocol for IPv6(DHCPv6)
- (3) RFC[4861] Neighbor Discovery for IP version 6

## Authors' Addresses

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

Qiongfang Ma  
China Mobile  
32, Xuanwumenxi Ave.  
Xicheng District, Beijing 100053  
China

Email: maqiongfang@chinamobile.com

