Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 2, 2015

G. Chen
China Mobile
W. Li
China Telecom
T. Tsou
J. Huang
Huawei Technologies
T. Taylor
PT Taylor Consulting

September 29, 2014

Analysis of NAT64 Port Allocation Methods for Shared IPv4 Addresses
draft-chen-sunset4-cgn-port-allocation-05

Abstract

This document enumerates methods of port assignment in Carrier Grade NATs (CGNs), focused particularly on NAT64 environments. A theoretical framework of different NAT port allocation methods is described. The memo is intended to clarify and focus the port allocation discussion and propose an integrated view of the considerations for selection of the port allocation mechanism in a given deployment.

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 2, 2015.

Table of Contents

1. Introduction

As a resdult of the depletion of IPv4 addresses, Carrier Grade NAT (CGN) has been adopted by ISPs to expand IPv4 spaces. CGN maps IP addresses from one address realm to another, relying upon the mechanism of multiplexing multiple subscribers' connections over a smaller number of shared IPv4 addresses to provide connectivity to

end hosts. [RFC6888] specifies a number of CGN requirements. A network-based NAT is implied by several approaches to IPv6 transition including DS-Lite [RFC6333], NAT64 ([RFC6145] and [RFC6146]), and NAT444. All of these would likely fall within the scope of the CGN requirements document [RFC6888].

The first part of this memo (Section 2) focusses on the topic of IPv6 migration. When NAPT is involved, Section 2 elaborates on the considerations for address sharing and particularly port assignment in the NAT64 environment, where IPv6-only nodes are connected to external dual-stack or IPv4 networks.

Section 3 looks more closely at dynamic bulk assignment of ports to individual subscriber sites, particularly as a means of log volume reduction. The proposals made in this section are applicable to the CGN environment in general, independently of the particular flavour of translation being used.

The considerations in this document do not apply where the CGN does only Network Address Translation (NAT) [RFC3022]. In this scenario, there is no concern about port assignment. Similarly, this document does not apply where encapsulation rather than translation is used as the IPv6 transition method.

2. Considerations For the Choice of Port Allocation Methods

For port allocations on NAT64, several aspects may have to be considered when selecting a suitable method. Here is a list of the potential considerations, which are covered in more detail below.

- specific features of port usage in a NAT64 environment;
- classification of different port allocation methods;
- port allocation to improve connectivity;
- port allocation to optimize log volume;
- port allocation to enhance security.

Both analysis and relevant experimental results are presented in the sub-sections that follow.

2.1. Port Consumption on NAT64

China Mobile did a test comparison of port consumption on NAT64 and NAT44. Top100 websites (referring to Alexa statistics) were assessed to evaluate status of port usage on NAT44 and NAT64 respectively.

China Mobile observed that the port consumption per session on NAT64 is roughly only half that on NAT44. 43 percent of top100 websites have AAAA records, therefore the NAT64 didn't have to assign ports to the traffic going to those websites. The results may be different if more services (e.g. game, web-mail, etc) are considered. But it is apparent that the effects of port saving on NAT64 will be amplified by increasing native IPv6 support.

Apart from the above observation, port allocation can be tuned according to the phase of IPv6 migration. As more content providers and services become available over IPv6, the utilization of NAT64 goes down since fewer destinations require translation progressing. Thus as IPv6 migration proceeds, it will be possible to relax the multiplexing ratio of IPv4 address sharing.

2.2. Classification of Port Allocation Models

This section lists several models to allocate the port information in NAT64 equipment. It also describes example cases for each allocation model.

2.2.1. Stateful vs. Stateless

- Stateful

  The stateful NAT can be implemented either by static address translation or dynamic address translation.

  In the case of static address assignment, a one-to-one address mapping for hosts between a IPv6 network address and an IPv4 network address is pre-configured on the NAT operation. This case normally occurs when a server is deployed in a IPv6 domain. The static configuration ensures stable inbound connectivity.

  Dynamic address assignment would periodically free the binding so that the global address could be recycled for later use. This increases the efficiency of usage of IPv4 addresses.

- Stateless

  Stateless NAT is performed in compliance with [RFC6145]. The public IPv4 address is required to be embedded in the IPv6 address. Thus the NAT64 can directly extract the address and has no need to record mapping states.

A promising usage of stateless NAT may appear in the data centre environment where IPv6 server pools receive inbound connections from IPv4 users externally [I-D.anderson-v6ops-siit-dc]. NAT usage in

other cases may be controversial. First off, the static one-to-one mapping does not address the issue of IPv4 depletion. Secondly, it introduces a dependency between IPv4 and IPv6 addressing. That creates new limitations since a change of IPv4 address will cause renumbering of IPv6 addresses.

2.2.2. Dynamic vs. Static

Port assignment can be dynamic (ports allocated on demand) or static (ports allocated as part of the configuration process).

- Dynamic assignment

  NAT64 normally uses dynamic assignment, since this achieves higher port utilization. Port allocations can be made with per-session or per-customer granularity. Per-session assignment is configured on the NAT64 by default since it maximizes port utilization. However, this can result in a heavy log volume that may have to be recorded for lawful interception systems. To mitigate that concern, the NAT64 may dynamically allocate a port range for each connected subscriber. This will significantly reduce log volume.

  A proper port-range configuration may have to take into account two considerations:

  A. The number of session initiations for each subscriber. A subscriber normally uses multiple applications simultaneously, e.g. map, online video or game. The number of concurrent sessions is essential to determine the number of ports the subscriber needs. The China Mobile study mentioned earlier observed that the average number of sessions consumed by one user's device was around 200 to 300 ports. Several devices may appear behind a CPE. Based on this observation, 1000 ports per subscriber household will provide enough room for multiple active users. Administrators should monitor usage to adjust this number if users are being limited by this number, or if usage is so low that fewer ports would be sufficient.

  B. Impacts on NAT64 capacity. Preassigned port ranges occupy memory even when there are unused ports. Therefore, the operator should be cautious about the impact of port-range reservation on the capacity for attempted concurrent sessions, especially in the case of a centralized NAT64 CGN serving numerous subscribers.

- Static assignment

Static assignment makes port reservations in bulk for each internal address before subscriber connection. The assigned ports can be in either a contiguous port range or a non-contiguous port range for the sake of defense against port-guessing attacks (see Section 3.2). Log recording may not be necessary due to the stable mapping relations. Considerations of the interaction between port-range allocation and capacity impact are also applicable in the case of static assignment. [I-D.donley-behave-deterministic-cgn] describes a deterministic algorithm to assign a port range for an internal IP address pool in a sequence.

2.2.3. Centralized vs. Distributed

There is an increasing need to connect NAT64 with downstream NAT46-capable devices to support IPv4 users/applications on an IPv6-only path. Several solutions have been proposed in this area, e.g., 464xlat [RFC6877], MAP-T [I-D.ietf-softwire-map-t] and 4rd [I-D.ietf-softwire-4rd]. Port allocation can be categorized as a centralized assignment on NAT64 or as a port delegation distributed to downstream devices (e.g, Customer Edge connected with NAT64).

- Centralized Assignment

  A centralized method makes port assignments once IP flows come to the NAT64. The allocation policy is enforced on a centralized point. Either a dynamic or static port assignment is made for received sessions.

- Distributed Assignment

  NAT64 can also delegate the pre-allocated port range to customer edge devices. That can be achieved through additional out-of-band provisioning signals (e.g., [I-D.ietf-pcp-port-set], [I-D.ietf-softwire-map-dhcp]). The distributed model normally is performed A+P style [RFC6346] for static port assignment. The NAT64 should also hold the corresponding mapping in order to validate port usage in the outgoing direction and route inbound packets. Delegated port ranges shift NAT64 port computations/ states into downstream devices. The detailed benefits of this approach are documented in [I-D.ietf-softwire-stateless-4v6-motivation].

2.3. Port Allocation Solutions

2.3.1. Other Transition Technologies

In other work, stateful NAT64 [RFC6146] uses bindings between IPv4 and IPv6 addresses that may be either static or dynamic. [RFC6146] describes a process where the dynamic binding is created by an outgoing packet, but it may also be created by other means such as a Port Control Protocol request (see Section 2.3.3). Lookin beyond NAT64 for the moment, DS-Lite [RFC6333] refers to the cautions in [RFC6269] but does not specify any port allocation method. Both technologies assume a centralized model.

The specifications for both transition methods thus allow implementations to use the proposals made in Section 3 (and [I-D.donley-behave-deterministic-cgn]).

2.3.2. Current Work On Stateless Transition Technologies

The port allocation solutions that are being specified at the time of writing of this document are all variations on the static distributed model, to minimize the amount of state that has to be held in the network. The proposals made in Section 3 do not apply to the current work in progress because that work has gone in another direction. That work includes:

- Light-weight 4over6 (LW4o6 [I_D.ietf-softwire-lw4over6]), which requires the CPE to be configured explicitly with the shared IPv4 address and port set it will use on the WAN side of its NAT44 function. The border router is configured with the same information, reducing the state it must hold from per-session to per-subscriber amounts.

- Mapping of Address and Port with Encapsulation (MAP-E [I-D.ietf-softwire-map]) and the experimental specifications Mapping of Address and Port with Translation (MAP-T [I-D.ietf-softwire-map-t]) and 4rd [I-D.ietf-softwire-4rd], already mentioned. These rely on an algorithmic embedding of WAN-side IPv4 address and assigned port set within the IPv6 prefix assigned to each CPE. Both the CPE and the border router must be configured with this information. However, the algorithm is designed to aggregate routing information such that the amount of state carried by the border router is of a lower order of magnitude than even the per-subscriber level.

MAP-E also supports a 1-1 mapping mode, where the IPv4 and IPv6 addresses assigned to a CPE are independent. This can be helpful in transition, but, as with LW4o6, raises the amount of state in the network back to the per-subscriber level.

For a packet destined to a host outside the MAP domain from which the packet originated: MAP-E and 4rd treat the packet as an IPv4 over IPv6 tunnel via the border router.

MAP-T uses stateless mapping in the sense of Section 2.2.1 by embedding the destination IPv4 address within the IPv6 address of the packet sent to the border router.

2.3.3. Port Control Protocol (PCP)

The Port Control Protocol (PCP, [RFC6887]) can be used to reserve a single port or a port set [I-D.ietf-pcp-port-set] for applications. It requires that the NAT be collocated with a PCP server function. PCP provides an out-of-band signalling mechanism for coordinating dynamic allocation of ports between hosts and the border router.

2.4. Specific Considerations

2.4.1. Log Volume Optimization

[RFC6269] has provided a thoughtful analysis on the issues of IP sharing. It points out that IP sharing may impact law enforcement since source address information will be lost during the translation. Network administrators have to log the mapping status for each connection in order to identify a specific user associated with an IP address in a particular time slot. The storage of log information may post a challenge to operators, since it requires additional resources and data inspection processes to identify users. For concrete details of what should be logged, see Section 3.1 of [I-D.ietf-behave-syslog-nat-logging]. The actual logging may use either IPFIX [RFC7011] or Syslog [RFC5424] depending on the operator's requirements.

It is desirable to reduce the volume of the logged information. Referring to the classification of port allocation methods given above, dynamic assignments can be managed on either a per-session or per-customer granularity. The coarser granularity will lead to lower log volume storage. A test was made by recording the log information from 200,000 subscribers in the Chinese network for 60 days. The volume of recorded information reached up to 42.5 terabytes with per-session logging in the raw format. The volume could be reduced to 10.6 terabytes with gzip format. Compared with that, it only occupied 40.6 gigabytes, three orders of magnitude smaller volume, with per-customer logging in the raw format. With static allocation, of course, no logs at all are required.

On the other hand, the lower logging volumes are associated with lower efficiency of port utilization. A port allocation based on

per-customer granularity has to retain vacant ports in order to avoid traffic overflow. The efficiency can be evaluated by port utilization rate, and will be even lower if the static port allocation method is used. Inactive users may also impact the efficiency.

Table 1 summarizes the test results using Syslog. The ports were pre-allocated to customers regardless of online or offline status.

Port Allocation Method	Log Granularity	Estimated Log Volume	Port Utilization
Dynamic NAPT	Per-session	42.5 terabytes	100%
Dynamic port-range	Per-customer	40.6 Gigabytes	75%
Deterministic NAT, MAP-T, 4rd	None	None	(60% * 75%) = 45%

Table 1: Estimated Log Volumes For 200,000 Users Over 60 Days

Note: 75% is the estimated port utilization ratio per active subscriber. 60% is the estimated ratio of active subscribers to the total number of subscribers.

The data shown in Table 1 roughly demonstrates the tradeoff between port utilization and log volume reduction. Administrators may consider the following factors to determine their own solution:

- average connectivity per customer per day;
- peak connectivity per day;
- the number of public IPv4 addresses available to the NAT64;
- application demands for specific ports;
- processing capabilities of the NAT64;
- tolerable log volume.

2.4.2. Connectivity State Optimization

It has been observed that port consumption is significantly increased once subscribers land on a web page for video on demand, an online game, or map services. In those cases, multiple TCP connections may be initiated to optimize the performance of data transmissions for video download and message exchange. Given the video traffic growth

trend, this likely presents a challenge for network operators who need to optimize connectivity states and avoid port depletion. Those optimizations may even affect the method of port-range allocation, because a subscriber is only allowed to use a pre-configured port resource.

Two optimizations may be considered:

- Reducing the TIME-WAIT state. The user's behavior normally correlates with system performance. It is rather common that users change video channels often. Investigations have shown that 60% of videos are watched for less than 20% of their duration. The user's access patterns may leave a number of the TIME-WAIT states. Therefore, acceleration of TIME-WAIT state transitions could increase the efficiency of port utilization. [RFC6191] defines a mechanism for reducing TIME-WAIT state by proposing TCP timestamps and sequence numbers.

  [I-D.penno-behave-rfc4787-5382-5508-bis] recommended applying [RFC6191] and PAWS (Protect Against Wrapped Sequence numbers, described in [RFC1323]) to NAT. This may also be a way to improve port utilization.

- Another possibility is to use Address-Dependent Mapping or Address and Port-Dependent Mapping [RFC4787] to increase port utilization. This feature has already been implemented on a vendor-specific basis. However, it should be noted that REQ-7 and REQ-12 in [RFC6888] may reduce the incentive to use anything but the Address-Independent Mapping behaviour recommended by [RFC4787].

2.4.3. Port Randomization

Port randomization is a feature to enhance the defense against hijacking of flows. [RFC6056] specifies that:

> "A NAPT that does not implement port preservation ([RFC4787], [RFC5382]) should obfuscate selection of the ephemeral port of a packet when it is changed during translation of that packet."

A NAPT based on per-session allocation normally follows this recommendation.

See Section 4 for a fuller discussion of port randomization.

3. Considerations For the Dynamic Assignment of Port-Ranges

3.1. Motivation

During the IPv6 transition period, large-scale NAT devices may be introduced, e.g. DS-Lite AFTR, NAT64. When a NAT device needs to set up a new connection for a given internal address behind the NAT, it needs to create a new mapping entry for the new connection, which will contain source IP address, source port or ICMP identifier, converted source IP address, converted source port, protocol (TCP/ UDP), etc.

For various reasons it is necessary to log these mappings. Some high performance NAT devices may need to create a large amount of new sessions per second. As seen in Section 2.4.1, if the logs are generated for each mapping entry, the log traffic could reach tens of megabytes per second or more, which would be a problem for log generation, transmission and storage. (The per-session volumes in Table 1 amount to 42 bytes per served subscriber per second. The volumes reported in the introduction to [I-D.donley-behave-deterministic-cgn] for U.S. users are even higher, around 58 bytes per second per subscriber served.)

[RFC6888], REQ-13, REQ-14, and REQ-15 deal explicitly with port allocation schemes and logging. However, it is recognized that these are conflicting requirements, requiring a tradeoff between the efficiency with which ports are used and the rate of generation of log records.

Allocating a range of N ports at once reduces the log volume by a factor of N, while also reducing port utilization by a factor which varies with the address sharing ratio and other configuration parameters. This provides a clear motivation to use dynamic allocation of port-ranges rather than individual ports when it is possible to do so while maintaining a satisfactory level of port utilization (and by implication, shared global IPv4 address utilization).

Dynamic allocation of port ranges may be used either as the sole strategy for port allocation on the NAPT, or as a supplement to an initial static allocation.

3.2. Implementation Issues -- Port Randomization and Port-Range Deallocation

When the user sends out the first packet, a port resource pool is allocated for the user, e.g., assigning ports 2001~2300 of a public IP address to the user's resource pool. Only one log should be

generated for this port block. When the NAT needs to set up a new mapping entry for the user, it can use a port in the user's resource pool and the corresponding public IP address. If the user needs more port resources, the NAT can allocate another port block, e.g., ports 3501~3800, to the user's resource pool. Again, just one log needs to be generated for this port block.

[I-D.bajko-pripaddrassign] takes this idea further by allocating non-contiguous sets of ports using a pseudorandom function. Scattering the allocated ports in this way provides a modest barrier to port guessing attacks. The use of randomization is discussed further in Section 4.

Suppose now that a given internal address has been assigned more than one block of ports. The individual sessions using ports within a port block will start and end at different times. If no ports in some port block are used for some configurable time, the NAT can remove the port block from the resource pool allocated to a given internal address, and make it available for other users. In theory, it is unnecessary to log deallocations of blocks of ports, because the ports in deallocated blocks will not be used again until the blocks are reallocated. However, the deallocation may be logged when it occurs to add robustness to troubleshooting or other procedures.

The deallocation procedure presents a number of difficulties in practice. The first problem is the choice of timeout value for the block. If idle timers are applied for the individual mappings (sessions) within the block, and these conform to the recommendations for NAT behaviour for the protocol concerned, then the additional time that might be configured as a guard for the block as a whole need not be more than a few minutes. The block timer in this case serves only as a slightly more conservative extension of the individual session idle timers. If, instead, a single idle timer is used for the whole block, it must itself conform to the recommendations for the protocol with which that block of ports is associated. For example, REQ-5 of [RFC5382] requires an idle timer expiry duration of at least 2 hours and 4 minutes for TCP. The suggestions made in Section 2.4.2 may be considered for reducing this time.

The next issue with port block deallocation is the conflict between the desire to randomize port allocation and the desire to make unused resources available to other internal addresses. As mentioned above, ideally port selection will take place over the entire set of blocks allocated to the internal address. However, taken to its fullest extent, such a policy will minimize the probability that all ports in any given block are idle long enough for it to be released.

As an alternative, it is suggested that when choosing which block to select a port from, the NAT should omit from its range of choice the block that has been idle the longest, unless no ports are available in any of the other blocks. The expression "block that has been idle the longest" designates the block in which the time since the last packet was observed in any of its sessions, in either direction, is earlier than the corresponding time in any of the other blocks assigned to that internal address. As [RFC6269] points out, port randomization is just one security measure of several, and the loss of randomness incurred by the suggested procedure is justified by the increased utilization of port resources it allows.

3.3. Issues Of Traceability

Section 12 of [RFC6269] provides a good discussion of the traceability issue. Complete traceability given the NAT logging practices proposed in this draft requires that the remote destination record the source port of a request along with the source address (and presumably protocol, if not implicit) [RFC6302]. In addition, the logs at each end must be timestamped, and the clocks must be synchronized within a certain degree of accuracy. Here is one reason for the guard timing on block release, to increase the tolerable level of clock skew between the two ends.

Where source port logging can be enabled, this memo strongly urges the operators to do so. Similarly, intrusion detection systems should capture source port as well as source address of suspect packets.

In some cases [RFC6269], a server may not record the source port of a connection. To allow traceability, the NAT device needs to record the destination IP address of a connection. As [RFC6269] points out, this will provide an incomplete solution to the issue of traceability because multiple users of the same shared public IP address may access the service at the same time. From the point of view of this draft, in such situations the game is lost, so to speak, and port allocation at the NAT might as well be completely dynamic.

The final possibility to consider is where the NAT does not do per-session logging even given the possibility that the remote end is failing to capture source ports. In that case, the port allocation strategy proposed in this section can be used. The impact on traceability is that analysis of the logs would yield only the list of all internal addresses mapped to a given public address during the period of time concerned. This has an impact on privacy as well as traceability, depending on the follow-up actions taken.

3.4. Other Considerations

[RFC6269] notes several issues introduced by the use of dynamic as opposed to static port assignment. For example, Section 12.2 of that document notes the effect on authentication procedures. These issues must be resolved, but are not specific to the dynamic port-range allocation strategy.

4. Security Considerations

The discussion which follows addresses an issue that is particularly relevant to the strategies described in Section 3 of this document. The security considerations applicable to NAT operation for various protocols as documented in, for example, [RFC4787] and [RFC5382] also apply to this proposal.

[RFC6056] summarizes the TCP port-guessing attack, by means of which an attacker can hijack one end of a TCP connection. One mitigating measure is to make the source port number used for a TCP connection less predictable. [RFC6056] provides various algorithms for this purpose.

As Section 3.1 of that RFC notes: "...provided adequate algorithms are in use, the larger the range from which ephemeral ports are selected, the smaller the chances of an attacker are to guess the selected port number." Conversely, the reduced range sizes proposed by the present document increase the attacker's chances of guessing correctly. This result cannot be totally avoided. However, mitigating measures to improve this situation can be taken both at port block assignment time and when selecting individual ports from the blocks that have been allocated to a given user.

At assignment time, one possibility is to assign ports as non-contiguous sets of values as proposed in [I-D.bajko-pripaddrassign]. However, this approach creates a lot of complexity for operations, and the pseudo randomization can create uncertainty when the accuracy of logs is important to protect someone's life or liberty.

Alternatively, the NAT can assign blocks of contiguous ports. However, at assignment time the NAT could attempt to randomize its choice of which of the available idle blocks it would assign to a given user. This strategy has to be traded off against the desirability of minimizing the chance of conflict between what [RFC6056] calls "transport protocol instances" by assigning the most-idle block, as suggested in Section 3. A compromise policy might be to assign blocks only if they have been idle for a certain amount of time whenever possible, and select pseudorandomly between the blocks available according to this criterion. In this case it is suggested

that the time value used be greater than the guard timing mentioned in Section 3, and that no block should ever be reassigned until it has been idle at least for the duration given by the guard timer.

Note that with the possible exception of cryptographically-based port allocations, attackers could reverse-engineer algorithmically-derived port allocations to either target a specific subscriber or to spoof traffic to make it appear to have been generated by a specific subscriber. However, this is exactly the same level of security that the subscriber would experience in the absence of CGN. CGN is not intended to provide additional security by obscurity.

While the block assignment strategy can provide some mitigation of the port guessing attack, the largest contribution will come from pseudo-randomization at port selection time. [RFC6056] provides a number of algoriths for achieving this pseudo-randomization. When the available ports are contained in blocks which are not in general consecutive, the algorithms clearly need some adaptation. The task is complicated by the fact that the number of blocks allocated to the user may vary over time. Adaptation is left as an exercise for the implementor.

5. IANA Considerations

This document makes no request of IANA.

6. Acknowledgements

This document is the result of a merger of the original draft-chen-sunset4-cgn-port-allocation and draft-tsou-behave-natx4-log-reduction. Version -02 of draft-chen contains the following acknowledgements:

> The author would like to thank Lee Howard and Simon Perreault for their helpful comments.
>
> Many thanks to Wesley George and Marc Blanchet encourage the author to continue this work.

The authors of draft-tsou-behave-natx4-log-reduction have their own thanks to give. Mohamed Boucadair reviewed the initial document and provided useful comments to improve it. Reinaldo Penno, Joel Jaeggli, and Dan Wing provided comments on the subsequent version that resulted in major revisions. Serafim Petsis provided encouragement to publication after a hiatus of two years.

The present version of the document benefited from further comments by Lee Howard.

7. References

7.1. Normative References

[RFC6056] Larsen, M. and F. Gont, "Recommendations for Transport-Protocol Port Randomization", BCP 156, RFC 6056, January 2011.

[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.

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

[RFC6888] Perreault, S., Yamagata, I., Miyakawa, S., Nakagawa, A., and H. Ashida, "Common Requirements for Carrier-Grade NATs (CGNs)", BCP 127, RFC 6888, April 2013.

7.2. Informative References

[I-D.anderson-v6ops-siit-dc]
Anderson, T., "SIIT-DC: Stateless IP/ICMP Translation for IPv6 Data Centre Environments (Work in progress)", September 2014.

[I-D.bajko-pripaddrassign]
Bajko, G., Savolainen, T., Boucadair, M., and P. Levis, "Port Restricted IP Address Assignment (expired Work in Progress)", April 2012.

[I-D.donley-behave-deterministic-cgn]
Donley, C., Grundemann, C., Sarawat, V., Sundaresan, K., and O. Vautrin, "Deterministic Address Mapping to Reduce Logging in Carrier Grade NAT Deployments (Work in progress)", January 2014.

[I-D.ietf-behave-syslog-nat-logging]
Chen, Z., Zhou, C., Tsou, T., and T. Taylor, "Syslog Format for NAT Logging (Work in Progress)", January 2014.

[I-D.ietf-pcp-port-set]
Sun, Q., Boucadair, M., Sivakumar, S., Zhou, C., Tsou, T., and S. Perrault, "Port Control Protocol (PCP) Extension for Port Set Allocation (Work in Progress)", July 2014.

[I-D.ietf-softwire-4rd]
Despres, R., Jiang, S., Penno, R., Lee, Y., Chen, G., and M. Chen, "IPv4 Residual Deployment via IPv6 - a Stateless Solution (4rd) (Work in Progress)", April 2014.

[I-D.ietf-softwire-map]
Troan, O., Dec, W., Li, X., Bao, C., Matsushima, S., Murakami, T., and T. Taylor, "Mapping of Address and Port with Encapsulation (MAP) (Work in Progress)", January 2014.

[I-D.ietf-softwire-map-dhcp]
Mrugalski, T., Troan, O., Dec, W., Farrer, I., Perrault, S., Bao, C., Yeh, L., and X. Deng, "DHCPv6 Options for configuration of Softwire Address and Port Mapped Clients (Work in Progress)", July 2014.

[I-D.ietf-softwire-map-t]
Li, X., Bao, C., Dec, W., Troan, O., Matsushima, S., and T. Murakami, "Mapping of Address and Port using Translation (MAP-T) (Work in progress)", February 2014.

[I-D.ietf-softwire-stateless-4v6-motivation]
Boucadair, M., Matsushima, S., Lee, Y., Bonness, O., Borges, I., and G. Chen, "Motivations for Carrier-side Stateless IPv4 over IPv6 Migration Solutions (Expired work in Progress)", November 2012.

[I-D.penno-behave-rfc4787-5382-5508-bis]
Penno, R., Perrault, S., Kamiset, S., Boucadair, M., and K. Naito, "Network Address Translation (NAT) Behavioral Requirements Updates (expired Work in Progress)", January 2013.

[I_D.ietf-softwire-lw4over6]
Cui, Y., Sun, Q., Boucadair, M., Tsou, T., Lee, Y., and I. Farrer, "Lightweight 4over6: An Extension to the DS-Lite Architecture (Work in Progress)", June 2014.

[RFC1323] Jacobson, V., Braden, B., and D. Borman, "TCP Extensions for High Performance", RFC 1323, May 1992.

[RFC3022] Srisuresh, P. and K. Egevang, "Traditional IP Network Address Translator (Traditional NAT)", RFC 3022, January 2001.

[RFC4787] Audet, F. and C. Jennings, "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP", BCP 127, RFC 4787, January 2007.

[RFC5382] Guha, S., Biswas, K., Ford, B., Sivakumar, S., and P. Srisuresh, "NAT Behavioral Requirements for TCP", BCP 142, RFC 5382, October 2008.

[RFC5424] Gerhards, R., "The Syslog Protocol", RFC 5424, March 2009.

[RFC6191] Gont, F., "Reducing the TIME-WAIT State Using TCP Timestamps", BCP 159, RFC 6191, April 2011.

[RFC6302] Durand, A., Gashinsky, I., Lee, D., and S. Sheppard, "Logging Recommendations for Internet-Facing Servers", BCP 162, RFC 6302, June 2011.

[RFC6333] Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", RFC 6333, August 2011.

[RFC6346] Bush, R., "The Address plus Port (A+P) Approach to the IPv4 Address Shortage", RFC 6346, August 2011.

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

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

[RFC7011] Claise, B., Trammell, B., and P. Aitken, "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information", STD 77, RFC 7011, September 2013.

Authors' Addresses

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

Email: phdgang@gmail.com

Weibo Li
China Telecom
109, Zhongshan Ave. West, Tianhe District
Guangzhou 510630
P.R. China

Email: mweiboli@gmail.com

Tina Tsou
Huawei Technologies
Bantian, Longgang District
Shenzhen 518129
P.R. China

Email: tina.tsou.zouting@huawei.com

James Huang
Huawei Technologies
Bantian, Longgang District
Shenzhen 518129
P.R. China

Email: James.huang@huawei.com

Tom Taylor
PT Taylor Consulting
Ottawa, Ontario
Canada

Email: tom.taylor.stds@gmail.com

Internet Engineering Task Force
Internet-Draft
Intended status: Best Current Practice
Expires: April 3, 2015

W. George
L. Howard
Time Warner Cable

September 30, 2014

IPv6 Support Within IETF work
draft-george-ipv6-support-03

Abstract

This document recommends that the IETF formally require its standards work to be IP version agnostic or to explicitly include support for IPv6, with some exceptions, to ensure that it is possible to operate without dependencies on IPv4.

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 3, 2015.

Table of Contents

1. Introduction

[RFC6540] gives guidance to implementers that in order to ensure interoperability and proper function after IPv4 exhaustion, IP-capable devices need to support IPv6, and cannot be reliant on IPv4, because global IPv4 exhaustion creates many circumstances where the use of IPv6 will no longer be optional. Since this is an IETF Best Current Practice recommendation, it is imperative that the results of IETF efforts enable implementers to follow that recommendation. This document provides recommendations and guidance as to how IETF itself should handle future work as it relates to Internet Protocol versions.

When considering support for IPv4 vs IPv6 within IETF work, the general goal is to provide tools that enable networks and applications to operate seamlessly in any combination of IPv4-only, dual-stack, or IPv6-only as their needs dictate. However, as the IPv4 to IPv6 transition continues, it will become increasingly difficult to ensure interoperability and backward compatibility with IPv4-only networks and applications. As IPv6 deployment grows, IETF will naturally focus on features and protocols that enhance and extend IPv6, along with continuing work on items that are IP version agnostic. New features and protocols will not typically be introduced for use as IPv4-only. However, as of this document's writing, there is no formal requirement for all IETF work to support IPv6, either implicitly by being network-layer agnostic or explicitly by having an IPv6-specific implementation.

1.1. 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].

2. IPv6-only operation

At this document's writing, IPv6 has seen significant deployment. Most of these deployments are dual-stack, with IPv4 and IPv6 coexisting on the same networks. However, dual-stack is a waypoint in the transition from IPv4 to IPv6. The eventual end state is networks and end points that are IPv6-only. Some operators may take a long time to turn off IPv4, if they ever do, but the IETF MUST ensure that its standards can be deployed by even the first operators to turn off IPv4. Problems (and solutions) need to be identified before they are encountered by the earliest adopters.

2.1. Functional Parity with IPv4

In order for IPv6-only operation to be realistic, IPv6 MUST have at least functional parity with IPv4. "Functional parity" means that any function that IPv4 enables MUST also be enabled by IPv6. This does not mean that every feature that exists in IPv4 will exist in IPv6; different features may enable the same function. For instance, IPv4 supports some features that are no longer in use. In some cases it has not been practical to remove them in IPv4, or even to declare them historic, but it is unnecessary to carry them forward into IPv6. IPv6 also eliminates the need for some features that exist in IPv4; no effort to create unneeded features is required. Functional parity does not mean that all functions in IPv6 must also be possible in IPv4. Indeed, with IPv6 becoming the predominant protocol, new functionality should be developed in IPv6, and IETF effort SHOULD NOT be spent retrofitting features into the legacy protocol.

2.2. IPv4 Sunset

Somewhat distinct from identifying the needed features for IPv6-only functional parity is the effort to identify what is necessary to disable or sunset IPv4 in a given network. Since many of the protocols in use today were designed to be fault-tolerant and very robust, actually removing them from a network once they are no longer needed is sometimes complex. Many implementations may not even have "off switches" because the assumption was that they would never be switched off in a normal network implementation. The Sunset4 Working Group was chartered to address these issues:

"The Working Group will point out specific areas of concern, provide recommendations, and standardize protocols that facilitate the graceful "sunsetting" of the IPv4 Internet in areas where IPv6 has been deployed. This includes the act of shutting down IPv4 itself, as well as the ability of IPv6-only portions of the Internet to continue to connect with portions of the Internet that remain IPv4-only. ... Disabling IPv4 in applications, hosts, and networks is new territory for much of the Internet today, and it is expected that problems will be uncovered including those related to basic IPv4 functionality, interoperability, as well as potential security concerns. The working group will report on common issues, provide recommendations, and, when necessary, protocol extensions in order to facilitate disabling IPv4 in networks where IPv6 has been deployed."

3. Requirements and Recommendations

Ongoing focus is required to ensure that future IETF work is capable of IPv6-only operation. This attention may take the form of IESG evaluation, individual document reviews, or future WG charters. Due to the existing operational base of IPv4, it is not realistic to completely bar further work on IPv4 within the IETF at this time, nor to formally declare it historic. Until the time when IPv4 is no longer in wide use and/or declared historic, the IETF needs to continue to update IPv4-only protocols and features for vital operational or security issues. Similarly, the IETF needs to complete the work related to IPv4-to-IPv6 transition tools for migrating more traffic to IPv6. As the transition to IPv6-capable networks accelerates, it is also likely that some changes may be necessary in IPv4 protocols to facilitate decommissioning IPv4 in a way that does not create unacceptable impact to applications or users. These sorts of IPv4-focused activities, in support of security, transition, and decommissioning, should continue, accompanied by problem statements based on operational experience. Generally the focus should move away from IPv4-only work.

> The IESG SHOULD review working group charters to ensure that work will be capable of operating without IPv4, except in cases of IPv4 security, transition, and decommissioning work.
>
> IETF SHOULD make updates to IPv4 protocols and features to facilitate IPv4 decommissioning
>
> IETF work SHOULD explicitly support IPv6 or SHOULD be IP version agnostic (because it is implemented above the network layer), except IPv4-specific transition or address-sharing technologies.
>
> IETF SHOULD NOT initiate new IPv4 extension technology development.

IETF work SHOULD function completely on IPv6-only nodes and networks, unless consensus exists that it is unnecessary to use a given feature or protocol on IPv6-only networks.

IETF SHOULD identify and update IPv4-only protocols and applications to support IPv6 unless consensus exists that it is unnecessary for a given feature or protocol.

4. Acknowledgements

Thanks to the following people for their comments: Jari Arkko, Ralph Droms, Scott Brim, Margaret Wasserman, Brian Haberman. Thanks also to Randy Bush, Mark Townsley, and Dan Wing for their discussion in IntArea WG at IETF 81 in Taipei, TW regarding transition technologies, IPv4 life extension, and IPv6 support.

5. IANA Considerations

This memo includes no request to IANA.

6. Security Considerations

This document generates no new security considerations because it is not defining a new protocol. As existing work is analyzed for its ability to operate properly on IPv6-only networks, new security issues may be identified.

7. References

7.1. Normative References

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

7.2. Informative References

[RFC6540] George, W., Donley, C., Liljenstolpe, C., and L. Howard, "IPv6 Support Required for All IP-Capable Nodes", BCP 177, RFC 6540, April 2012.

Authors' Addresses

Wesley George
Time Warner Cable
13820 Sunrise Valley Drive
Herndon, VA 20171
US

Phone: +1 703-561-2540
Email: wesley.george@twcable.com

Lee Howard
Time Warner Cable
13820 Sunrise Valley Drive
Herndon, VA 20171
US

Phone: +1-703-345-3513
Email: lee.howard@twcable.com

ï»¿

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

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

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

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

Abstract

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

Status of This Memo

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.

Table of Contents

1. Introduction

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

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

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

2. Related Work

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

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

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

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

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

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

- Consideration of whether the specification can operate in an environment without IPv4.
- Guidance on the use of 32-bit identifiers that are commonly populated by IPv4 addresses.

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

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

3. Remotely Disabling IPv4

3.1. Indicating that IPv4 connectivity is unavailable

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

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

3.2. Disabling IPv4 in the LAN

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

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

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

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

4. Client Connection Establishment Behavior

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

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

5. Disabling IPv4 in Operating System and Applications

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

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

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

6. On-Demand Provisioning of IPv4 Addresses

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

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

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

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

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

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

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

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

7. IPv4 Address Literals

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

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

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

8. Managing Router Identifiers

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

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

9. IANA Considerations

None.

10. Security Considerations

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

11. Acknowledgements

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

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

12. Informative References

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Annex A. Solution Ideas

A.1. Remotely Disabling IPv4

A.1.1. Indicating that IPv4 connectivity is unavailable

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

A.1.2. Disabling IPv4 in the LAN

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

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

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

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

A.2. Client Connection Establishment Behavior

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

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

A.3. Disabling IPv4 in Operating System and Applications

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

A.4. On-Demand Provisioning of IPv4 Address

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

A.5. Managing Router Identifiers

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

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

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

Authors' Addresses

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

Email: liushucheng@huawei.com

Weiping Xu
Huawei Technologies
Bantian, Longgang District
Shenzhen 518129
China

Email: xuweiping@huawei.com

Cathy Zhou
Huawei Technologies
Bantian, Longgang District
Shenzhen 518129
China

Email: cathy.zhou@huawei.com

Tina Tsou
Philips Lighting
United States of America

Email: tina.tsou@philips.com

Simon Perreault
Jive Communications
Quebec, QC
Canada

Email: sperreault@jive.com

Peng Fan
Beijing
China

Email: fanp08@gmail.com

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

Email: gurong_cmcc@outlook.com

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

Email: xiechf.bri@chinatelecom.cn

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

Email: chengying10@chinaunicom.cn