Network Working Group
Internet-Draft
Intended status: Informational
Expires: August 17, 2014

Q. Sun
China Telecom
W. Liu
C. Zhou
Huawei Technologies
G. Leclanche
Viagenie

February 13, 2014

Problem Statement for Openv6 Scheme
draft-sun-openv6-problem-statement-01

Abstract

This document assesses the variety and complexity of IPv6 deployments, and proposes a new space of study to simplify the enablement of new IPv6 applications on an existing network. The document evaluates the identified technical gaps as well.

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 August 17, 2014.

Table of Contents

1. Introduction 2
2. Terminology 3
3. Problem Extent and Existing Work 3
3.1. Variety of IPv6 deployment technologies 3
3.2. Complexity of IPv6 operation 5
3.2.1. End-to-End Network Management 5
3.2.2. Open Network Business Capabilities 6
3.3. Existing evaluations of the IPv6 Transition Landscape . . 7
4. Alternative Approach to IPv6 applications enablement 8
5. Existing protocols and methods for the alternate approach . . 9
6. Missing protocols and methods for the alternate approach . . 9
6.1. Dynamic devices forwarding table configuration 9
6.2. Address Management 9
7. Security Considerations 9
7.1. Source Address Validation and Traceback with Openv6 . . . 10
8. IANA Considerations 10
9. Authors 10
10. Acknowledgements 10
11. References 10
11.1. Normative References 10
11.2. Informative References 10
Authors' Addresses 12

1. Introduction

The exhaustion of the IPv4 address space has been a practical problem that providers are facing today. Network address migration to IPv6 is ongoing or upcoming throughout the world. However, IPv6 activation requires costly end-to-end network upgrades and different network scenarios will co-exist during IPv6 transition. In addition, the technologies deployed for the transition are suppose to be obsoleted once the transition is completed.

This document proposes a new approach to deploy and operate IPv6 applications on a network, whether related to transition technologies or purely native ones. Such a technology would allow to continue using the same equipments and operational practices for various deployment scenarios.

2. Terminology

3. Problem Extent and Existing Work

3.1. Variety of IPv6 deployment technologies

The IPv6 transition period contains three stages for IP Networks: IPv4-only, dual-stack and IPv6-only. The networks should support both IPv4 services and IPv6 services during each stage.[One-vision-for-IPv6]

There are multiple IPv6 transition technologies for different network scenarios (e.g. IPv4 network for IPv4/IPv6 user access, IPv6 network for IPv4/IPv6 user access, IPv4 servers for IPv6 visitors, etc.). Different network scenarios will co-exist during the IPv6 transition period, which means the devices implementing the IPv6 transition technology should support the array of technologies, or there has to be as many devices as technologies used in a given network. The following scenarios below will happen during the IPv6 transition period :

Scenario 1: An IPv6 host visits IPv6 servers via an IPv4 access network

Scenario 2: An IPv4 host visits IPv4 servers via an IPv4 NAT Dual-stack network

Scenario 3: An IPv6 host visits IPv6 servers via an IPv6 network

Scenario 4: An IPv4 host visits IPv4 servers via an IPv6 access network

Scenario 5: IPv4 host and IPv6 host interaction

Different transition mechanisms may have different impacts on user experience. For example, DS-Lite would have some impact due to address sharing compared to 6rd mechanisms, and NAT64 would have extra impact due to ALG issue. An operator having a diverse customer base might have to deploy different transition technologies for a given scenario depending on the required user experience. This implies that it is useful to support multiple transition mechanisms in the same area, and preferably on the same transition devices.

Another use case is that multiple scenarios may exist in the same stage. For example, if there are both IPv6-only devices and IPv4-only host in the same area with limited public IPv4 address, both NAT64 and NAT44 (or DS-Lite) are required to achieve IPv4 service connectivity.

The current implementations normally use a separate instance for each mechanism, and additional policies need to be applied when running multiple mechanisms in one device. Some have a limitation on the number of policies that can be configured in one device, while some have restrictions regarding the resource occupation (e.g. one transition instance will use a static amount of memory). The major challenges of IPv6 deployment mainly lie in two aspects:

The need to implement different IPv6 transition technologies in the same hardware and the need to support this by upgrading network devices as little as possible.

The need to hop over legacy infrastructures which are not IPv6 enabled, costly or impossible to upgrade.

The issues are:

1. How to support multiple transition mechanisms in a cost-efficient and flexible way ?

2. How to easily identify the transition type of different subscribers ?

A random operator will most likely not go through each scenario one by one. For example, some operators may start from scenario 1, and some may start directly from scenario 2 or scenario 4. However, since the target scenario is the IPv6-only access network, a single operator will be confronted to multiple scenarios on the long term.

In such a case, the operator should either upgrade existing devices to support new features, or replace them with new ones. In particular, when the operator's network consists of devices from different vendors, it is difficult to guarantee that all the legacy devices can be upgraded at the same time. This is costly and operationally complicated.

We call Transition Data Plane (TDP) the data forwarding plane of the operator network during the whole transition period. Issues that can be identified to improve the situation are:

1. How to manipulate Transition Data Plane with different modes?

2. How to identify the capabilities of different transition devices ?

3. How does the Transition Data Plane identify different modes in the unified platform ?

3.2. Complexity of IPv6 operation

3.2.1. End-to-End Network Management

3.2.1.1. Scattered Address Pool Management

When operators are facing the IPv4 address shortage problem, the remaining IPv4 address pools are usually quite scattered. It is quite complicated for an operator to manage scattered address pools in many transition devices. The situation will become even worse when multiple transition mechanisms in the same device need to be configured with different address pools. Besides, the occupation of the address pools may vary during different transition periods: when there is not many IPv6-enabled services and IPv6-enabled devices, IPv4 traffic will still represent a great portion of the total traffic, while in the later stage of IPv6 transition, IPv4 traffic will decrease and the amount of allocated IPv4 addresses may decrease as well, depending on customer requirements.

A solution could be to manage the address pools centrally. Different transition mechanisms can require the address pools on-demand. For example, when one transition mechanism is running out of the current address pools, it may request a additional address pool. It can also release the address pools that it is not using any longer. In this way, operators do not need to configure the address pools one by one manually and it also helps using the address pools more efficiently.

Fixing this problem implies solving those issues:

1. How to configure the address pools for different mechanisms ?

2. How to collect the current status of address pool usage ?

3.2.1.2. Source Address Validation and Traceback with Openv6

It has been long known the IPv4/IPv6 transition makes the tracking and validating of source IP address challenging. Whenever an IPvX packet is translated into an IPvY packet, a major change happens to the IP packet, which brings new issues:

1. How to track the origin of the IPvY packet which is actually in the IPvX world?

2. How to validate the IPvX packet at the edge of the IPvY world to prevent possible spoofing?

3. How to protect the IPvY address from being spoofed in the IPvY world?

SAVI[RFC7039] defines the source address validation solutions for both IPv4 and IPv6, but doesn't cover the scenario where an IPv4/IPv6 transition technology is used in the network. Currently designing a solution for the transition scenario is not an easy task. There are two main challenges:

1. the diversity of IPv4/IPv6 transition mechanisms. There have been a number of transition mechanism. Moreover, new transition mechanisms may be standardized in the future. It would be complex for a SAVI solution to understand each transition mechanism. An unified abstraction of the transition mechanisms (for example, an abstract Openv6 Transition Data Plan (TDP)) and a set of unified open interfaces should be provided by Openv6 to the SAVI solution for the transition scenario. Then the SAVI solution could know the correspondences between the two IP protocols without having to inspect each packet or keep heavy state locally. The SAVI solution can then generate filtering rules and process tracking.

2. the inflexibility of SAVI. Currently SAVI solutions are tightly associated with address assignment mechanisms. It should be noted that each IPv4/IPv6 transition mechanism actually introduce a new mechanism to assign valid IPv4/IPv6 addresses. Based on the current model of SAVI, the SAVI solution for the transition scenario should be able to track the address translation in all the transition mechanism. Such a SAVI solution is heavy and costly for switches. The SAVI solution should introduce flexibility in rule generation similarly as Openv6, which offloading the complexity from network devices to a controller.

3.2.2. Open Network Business Capabilities

3.2.2.1. Dynamic QoS guarantee in IPv6 transition period

Traditionally, almost all bandwidth on the Internet is shared, or with a pre-configured QoS class. However, since the QoS requirements by different applications are not always the same, the subscribers should either waste money by paying for a higher bandwidth service, or can not get qualified service when needed. Therefore, currently, operators are tending to provide more dynamic QoS guarantee for subscribers so that they may apply for a higher bandwidth on-demand when they needed, or specific QoS guarantee can be applied for a certain amount of applications. In this case, the QoS adjustment platform is needed to pass the QoS adjustment request from subscribers or application servers dynamically.

In IPv6 transition period, the situation will become more complicated. When CGNs are introduced in the network, ip address and port will change during the translation or tunnelling process. For

some solutions, e.g. NAT444, DS-Lite, etc., the mappings might be different for different sessions.

In this case, the QoS adjustment platform should have the ability to pass and acquire QoS requirements for certain mappings in the CGNs. Therefore, more flexibility should be introduced in the network to load the dynamic QoS requests to the forwarding devices, no matter whether it is a tunnelling or translating mapping.

3.2.2.2. Coordinated NAT translation

Traditionally, most peer-to-peer applications would deploy relays by their own to achieve NAT traversal. They may use different kinds of ways e.g. TURN, STUN, or use some private protocols for their own purpose. It would not only cost a lot for applications deploy multiple relays, but also introduces a lot of complexity for newly emerging applications. In addition, in IPv6 transition period, there would be more CGNs than before which might make it more difficult for applications to achieve NAT traversal.

However, when operators have deployed some kinds of CGNs in their network, it is reasonable for operators to provide NAT traversal service for third-party applications so that the applications do not need to deploy the relays by their own. For example, the third-party application may require the CGN with the transport address, reflect address, etc., and then choose the one to use for the specific NAT situation. It can also be applied when IPv6 client communicates with iPv4 client with similar procedure. In this case, a centralized controller is needed to acquire the requests from third-party applications and form the specific mappings for them.

3.3. Existing evaluations of the IPv6 Transition Landscape

This paragraph references work done at the IETF or to describe the complex landscape of transition technologies.

The different network environments (architecture, scale, services deployed, varying IP traffic) cause a variety of IPv6 transition technologies for different operators. This section analyses the current and future coexistence of IPv6 transition technologies situation as well as the issues behind it.

Since IPv6 was proposed, there have been a couple of RFCs and on-going documents in IETF, as listed in the table below.

status	number	documents
RFC	8 or more	[RFC5571], [RFC6333], [RFC6674], [RFC5969], [RFC6219], [RFC6535], [RFC6654], [RFC6145], ...
WG draft	6 or more	[I-D.ietf-softwire-4rd], [I-D.ietf-softwire-map], [I-D.ietf-softwire-map-t], [I-D.ietf-softwire-public-4over6], [I-D.ietf-softwire-lw4over6], [I-D.ietf-v6ops-464xlat], ...
Active draft	several	...

Table 1: A Table of IPv6 Transition Technologies @ IETF

The situation described above depicts the difficulty of selecting appropriate IPv6 transition technologies for the carriers. Moreover, according to [SD-NAT], there are multiple stages during the whole IPv6 transition period, and a variety of technologies and equipments are used during different IPv6 transition stages. To protect the user experience and the early investment, an operator will not upgrade its network directly to the final stage of IPv6 transition. During different IPv6 transition stages, an operator needs different technologies in different stages. Thus, a method that is able to implement different IPv6 transition technologies in the same hardware is crucial, to avoid repeated investments.

4. Alternative Approach to IPv6 applications enablement

Finally an IP Network is simply an interconnection of various IPv4- and IPv6-aware devices over some transport. From a payload point of view, there is no need to wonder how the packet got to the destination (security aspects are reserved). Removing the complexity of the transport from the IP-aware devices, by simply considering it as a hop-by-hop "encapsulation" would simplify some situations and bring more flexibility for new applications.

The alternative approach proposed here is to put the IPv6 forwarding rules into the devices by a dynamic configuration protocol like Netconf, depending on the application requirements. Those forwarding rules could for example require a change of encapsulation (e.g. from IPv6oEthernet to IPv6oIPv4oEthernet), or an IP protocol change (e.g. apply a NAT64 translation). A central management server would be able to coordinate this configuration and push it adequately on the forwarding devices.

Today, the configuration of these encapsulation or translations is done manually and is not controlled in a coordinated and standard way. The goal of the application-based approach is to allow the operator to have both the flexibility and full control on what technologies have to be used and when to help with its IPv6 transition process.

5. Existing protocols and methods for the alternate approach

The proposed approach would have impact on layer 3, and maybe 4. Hence there is no need to change anything to Layer 1-2 protocols and techniques.

Higher layer applications are not impacted either as the network forwarding is transparent to them.

The proposed approach requires a dynamic configuration protocol for network devices, to update the forwarding table accordingly. Protocols like Netconf (add ref) or Openflow (add ref) are already existing to achieve this goal. Thanks to their openness, they can easily be extended to support it.

6. Missing protocols and methods for the alternate approach

The authors have identified some missing pieces to be able to use the technology in a fully standard way.

6.1. Dynamic devices forwarding table configuration

The IETF standard for devices configuration is [RFC6241], the NETCONF Protocol. So it may be suitable for the forwarding table configuration of the openv6 devices and the address management in [section 6.2], with some modifications of the code. However, Netconf is not able to support the packet report from the device to the controller/applications, which may need extensions of the protocol.

6.2. Address Management

Having a centralized way to manage addresses requires an efficient protocol to request and allocate them. Among the possible solutions, Netconf or Radius could be extended.

7. Security Considerations

7.1. Source Address Validation and Traceback with Openv6

A easy-to-use solution for Source Address Validation would increase the safety of networks. If operators have an efficient and low cost unified solution for this problem for both IPv4 and IPv6 and the transition itself, they would be more incline to implement it and therefore the security of networks as a whole would improve.

8. IANA Considerations

This document has no actions for IANA.

9. Authors

Credits and Thanks

10. Acknowledgements

Reference previous work.

11. References

11.1. Normative References

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

11.2. Informative References

[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)", draft-ietf-softwire-4rd-07 (work in progress), October 2013.

[I-D.ietf-softwire-lw4over6]
Cui, Y., Qiong, Q., Boucadair, M., Tsou, T., Lee, Y., and I. Farrer, "Lightweight 4over6: An Extension to the DS-Lite Architecture", draft-ietf-softwire-lw4over6-06 (work in progress), February 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)", draft-ietf-softwire-map-t-05 (work in progress), February 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)", draft-ietf-softwire-map-10 (work in progress), January 2014.

[I-D.ietf-softwire-public-4over6]
Cui, Y., Wu, J., Wu, P., Vautrin, O., and Y. Lee, "Public IPv4 over IPv6 Access Network", draft-ietf-softwire-public-4over6-10 (work in progress), July 2013.

[I-D.ietf-v6ops-464xlat]
Mawatari, M., Kawashima, M., and C. Byrne, "464XLAT: Combination of Stateful and Stateless Translation", draft-ietf-v6ops-464xlat-10 (work in progress), February 2013.

[One-vision-for-IPv6]
Mark Townsley, "One vision for IPv6", .

[RFC5571] Storer, B., Pignataro, C., Dos Santos, M., Stevant, B., Toutain, L., and J. Tremblay, "Softwire Hub and Spoke Deployment Framework with Layer Two Tunneling Protocol Version 2 (L2TPv2)", RFC 5571, June 2009.

[RFC5969] Townsley, W. and O. Troan, "IPv6 Rapid Deployment on IPv4 Infrastructures (6rd) -- Protocol Specification", RFC 5969, August 2010.

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

[RFC6219] Li, X., Bao, C., Chen, M., Zhang, H., and J. Wu, "The China Education and Research Network (CERNET) IVI Translation Design and Deployment for the IPv4/IPv6 Coexistence and Transition", RFC 6219, May 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.

[RFC6654] Tsou, T., Zhou, C., Taylor, T., and Q. Chen, "Gateway-Initiated IPv6 Rapid Deployment on IPv4 Infrastructures (GI 6rd)", RFC 6654, July 2012.

[RFC6674] Brockners, F., Gundavelli, S., Speicher, S., and D. Ward, "Gateway-Initiated Dual-Stack Lite Deployment", RFC 6674, July 2012.

[RFC6674] Brockners, F., Gundavelli, S., Speicher, S., and D. Ward, "Gateway-Initiated Dual-Stack Lite Deployment", RFC 6674, July 2012.

[RFC7039] Wu, J., Bi, J., Bagnulo, M., Baker, F., and C. Vogt, "Source Address Validation Improvement (SAVI) Framework", RFC 7039, October 2013.

[SD-NAT] Alain Durand, "SD-NAT", <http://www.ietf.org/proceedings/82/slides/behave-10.pdf>.

Authors' Addresses

Qiong Sun
China Telecom
No.118 Xizhimennei street, Xicheng District
Beijing 100035
P.R. China

Email: sunqiong@ctbri.com.cn

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

Email: liushucheng@huawei.com

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

Email: cathy.zhou@huawei.com

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

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