Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 9, 2012

Y. Cui
J. Wu
P. Wu
Tsinghua University
C. Metz
Cisco Systems, Inc.
O. Vautrin
Juniper Networks
Y. Lee
Comcast
July 8, 2011

Public IPv4 over Access IPv6 Network
draft-cui-softwire-host-4over6-06

Abstract

This draft proposes a mechanism for bidirectional IPv4 communication between IPv4 Internet and end hosts or IPv4 networks sited in IPv6 access network. This mechanism follows the softwire hub and spoke model and uses IPv4-over-IPv6 tunnel as basic method to traverse IPv6 network. By allocating public IPv4 addresses to end hosts/networks in IPv6, it can achieve IPv4 end-to-end bidirectional communication between these hosts/networks and IPv4 Internet. This mechanism is an IPv4 access method for hosts and IPv4 networks sited in IPv6.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

Global IPv4 addresses are running out fast. Meanwhile, the demand for IP address is still growing and may even burst in potential circumstances like "Internet of Things". To satisfy the end users, operators have to push IPv6 to the front, by building IPv6 networks and providing IPv6 services.

When IPv6-only networks are widely deployed, users of those networks will probably still need IPv4 connectivity. This is because part of Internet will stay IPv4-only for a long time, and network users in IPv6-only networks will communicate with network users sited in the IPv4-only part of Internet. This demand could eventually decrease with the general IPv6 adoption.

Network operators should provide IPv4 services to IPv6 users to satisfy their demand, usually through tunnels. This type of IPv4 services differ in provisioned IPv4 addresses. If the users can't get public IPv4 addresses (e.g., new network users join an ISP which don't have enough unused IPv4 addresses), they have to use private IPv4 addresses on the client side, and IPv4-private-to-public translation is required on the carrier side, as is described in Dual-stack Lite[I-D.ietf-softwire-dual-stack-lite]. Otherwise the users can get public IPv4 addresses, and use them for IPv4 communication. In this case, translation on the carrier side won't be necessary. The network users and operators can avoid all the issues raised by translation, such as ALG, NAT traversal, state maintenance, etc. Note that this "public IPv4" situation is actually quite common. There're approximatively 2^32 network users who are using or can potentially get public IPv4 addresses. Most of them will switch to IPv6 sooner or later, and will require IPv4 services for a significant period after the switching. This draft focuses on this situation, i.e., to provide IPv4 access for users in IPv6 networks, where public IPv4 addresses are still available for allocation.

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. Terminology

Public 4over6: Public 4over6 is the mechanism proposed by this draft. Generally, Public 4over6 supports bidirectional communication between IPv4 Internet and IPv4 hosts or local networks in IPv6 access network, by leveraging IPv4-in-IPv6 tunnel and public IPv4 address allocation.

4over6 initiator: in Public 4over6 mechanism, 4over6 initiator is the IPv4-in-IPv6 tunnel initiator located on the user side of IPv6 network. The 4over6 initiator can be either a dual-stack capable host or a dual-stack CPE device. In the former case, the host has both IPv4 and IPv6 stack but is provisioned with IPv6 access only. In the latter case, the CPE has both IPv6 interface for access to ISP network and IPv4 interface for local network connection; hosts in the local network can be IPv4-only.

4over6 concentrator: in Public 4over6 mechanism, 4over6 concentrator is the IPv4-in-IPv6 tunnel concentrator located in IPv6 ISP network. It’s a dual-stack router which connects to both the IPv6 network and IPv4 Internet.

4. Deployment scenario

4.1. Scenario and requirements

The general scenario of Public 4over6 is shown in Figure 1. Users in an IPv6 network take IPv6 as their native service. Some users are end hosts which face the ISP network directly, while others are local networks behind CPEs, such as a home LAN, an enterprise network, etc. The ISP network is IPv6-only rather than dual-stack, which means that ISP can’t provide native IPv4 access to its users; however, it’s acceptable that one or more routers on the carrier side become dual-stack and get connected to IPv4 Internet. So if network users want to connect to IPv4, these dual-stack routers will be their "entrances".

```
                    +-------------------------+
                    |    IPv6 ISP Network     |
                 +------+                     |
                 |host: |                     |
                 |initi-|                     |
                 |ator  |=================+-------+   +-----------+
                 +------+                 |4over6 |   |   IPv4    |
                    |      IPv4-in-IPv6   |Concen-|---| Internet  |
   +----------+  +------+                 |trator |   |           |
   |local IPv4|--|CPE:  |=================+-------+   +-----------+
   | network  |  |initi-|                     |
   +----------+  |ator  |                     |
                 +------+                     |
                    |                         |
                    +-------------------------+
```

Figure 1 Public 4over6 scenario

Before getting into any technical details, the communication requirements should be stated. The first one is that, 4over6 users require IPv4-to-IPv4 communication with the IPv4 Internet. An IPv4 access service is needed rather than an IPv6-to-IPv4 translation service. (IPv6-to-IPv4 communication is out of the scope of this draft.)

Second, 4over6 users require public IPv4 addresses rather than private addresses. Public IPv4 address means there’s no IPv4 CGN along the path, so the acquired IPv4 service is better. In particular, some hosts may be application servers, public address works better for reasons like straightforward access, direct DNS registration, no stateful mapping maintenance on CGN, etc. For the direct-connected host case, each host should get one public IPv4 address. For the local IPv4 network case, the CPE can get a public IPv4 address and runs an IPv4 NAT for the local network. Here a local NAT is still much better than the situation that involves a CGN, since this NAT is in local network and can be configured and managed by the users.

Third, translation is not preferred in this scenario. If this IPv4-to-IPv4 communication is achieved by IPv4-IPv6 translation, it'll need double translation along the path, one from IPv4 to IPv6 and the other from IPv6 back to IPv4. This would be quite complicated, especially in addressing. Contrarily a tunnel can achieve the IPv4-over-IPv6 traversing easily. That's the reason this draft follows the hub and spoke softwire model.

Moreover, the ISP probably would like to keep their IPv4 and IPv6 addressing and routing separated when provisioning IPv4 over IPv6. Then the ISP can manage the native IPv6 network more easily and independently, and also provision IPv4 in a flexible, on-demand way. The cost is that the concentrator needs to maintain per-user address mapping state, which would be described in detail.

4.2. Use cases

Public 4over6 can be applicable in several practical cases. The first one is that ISPs which still own enough IPv4 addresses switch to IPv6. The ISPs can deploy public 4over6 to preserve IPv4 service for the customers. This case is actually quite common. The majority of the wired end users today get Internet access with public IPv4 address. When their ISPs switch to IPv6, these users can still use the same amount of IPv4 addresses for IPv4 access. Public 4over6 can leveraging these addresses and offer tunneled IPv4 access.

The second case is ISPs which don't have enough IPv4 addresses any more switch to IPv6. For these ISPs, dual-stack lite is so far the most mature solution to provision IPv4 over IPv6. In dual-stack lite, end users use private IPv4 addresses, experience a 44CGN and hence some service degradation. As long as the end users use public IPv4 addresses, all CGN issues can be avoided and the IPv4 service can be full bi-directional. In other words, Public 4over6 can be deployed along with DS-lite, to provide a value-added service. Common users adopt DS-lite to communicate with IPv4 while high-end users adopt Public 4over6. The two mechanisms can actually be coupled easily.

There is also a special situation in the second case that the end users are IPv4 application servers. In this situation, public address brings significant convenience. The DNS registration can be direct using dedicated address; the access of application clients can be straightforward with no translation; there's no need to reserve and maintain address mapping on the CGN, and no well-known port collision will come up. So it's better to have servers adopt Public 4over6 for IPv4 access when they're located in IPv6 network.

Following the principle of Public 4over6, it's also possible to achieve address multiplexing and save IPv4 addresses. There're already efforts on this subject, see [I-D.cui-softwire-b4-translated-ds-lite] and [I-D.sun-v6ops-laft6]. The basic idea is that instead of allocating a full IPv4 address to every end user, the ISP can allocate an IPv4 address with restricted port range to every end user.

Besides, the draft would like to be explicit about the scope of direct-connected host case and CPE case. The host case is clear: the host is directly connected to IPv6 network, but the protocol stack on the host support IPv4 too. As to the CPE case, this draft would like to only focus on the case that the local network behind the CPE is private IPv4. If the users want to run public IPv4 into the local network, then they can either run dual-stack in the local network and turn into host case(likely home LAN situation), or they can acquire address blocks from the ISP and build configured tunnel or softwire mesh[RFC5565] with the ISP network(likely enterprise network situation). TC can be implemented to be compatible with the latter case too, though.

5. Public 4over6 Mechanism

5.1. Address allocation and mapping maintenance

Public 4over6 can be generally considered as IPv4-over-IPv6 hub and spoke tunnel using public IPv4 address. Each 4over6 initiator will use public IPv4 address for IPv4-over-IPv6 communication. As is described above, in the host initiator case, every host will get one IPv4 address; in the CPE case, every CPE will get one IPv4 address, which will be shared by hosts behind the CPE. The key problem here is IPv4 address allocation over IPv6 network, from ISP device(s) to separated 4over6 initiators.

There're two possibilities here. One is DHCPv4 over IPv6, and the other is static configuration. DHCPv4 over IPv6 is achieved by performing DHCPv4 on IPv4-in-IPv6 tunnel between ISP device and 4over6 initiators. There do exist the DHCP encapsulation issue on server side, see details and solutions in [I-D.cui-softwire-dhcp-over-tunnel]. As to static configuration, 4over6 users and the ISP operators should negotiate beforehand to authorize the IPv4 address. Application servers usually falls into this case. Public 4over6 supports both address allocation manners. Actually, it is transparent to address allocation methods.

Along with IPv4 address allocation, Public 4over6 should maintain the IPv4-IPv6 address mappings on the concentrator. In this type of address mapping, the IPv4 address is the public IPv4 address allocated to a 4over6 initiator, and the IPv6 addresses is the initiator's IPv6 address. This mapping is used to provide correct encapsulation destination address for the concentrator.

The initiator sends "pinhole" packets to the concentrator periodically, to install and renew the address mapping. A pinhole packet is an IPv4-in-IPv6 packet, which uses the concentrator's IPv6 address as destination IPv6 address, the initiator's IPv6 address as source IPv6 address, and the initiator's IPv4 address as source IPv4 address. When the concentrator receives such a packet, it'll resolve the IPv4 and IPv6 address information from the packet and trigger the mapping. Since any IPv4-in-IPv6 data packet from the initiator contains these exact informations, it can also serve as pinhole packet. Then dedicated pinhole packets are sent out when there's no data packets. Another possible way to maintain the address mapping is to run PCP[I-D.ietf-pcp-base] while extending the protocol to support applying for a full address. The following sections describe the mechanism with the pinhole method.

5.2. 4over6 initiator behavior

4over6 initiator has an IPv6 interface connected to the IPv6 ISP network, and a tunnel interface to support IPv4-in-IPv6 encapsulation. In CPE case, it has at least one IPv4 interface connected to IPv4 local network.

4over6 initiator should learn the 4over6 concentrator's IPv6 address beforehand. For example, if the initiator gets its IPv6 address by DHCPv6, it can get the 4over6 concentrator's IPv6 address through a DHCPv6 option[I-D.ietf-softwire-ds-lite-tunnel-option].

5.2.1. Host initiator

When the initiator is a direct-connected host, it assigns the allocated public IPv4 address to its tunnel interface. The host uses this address for IPv4 communication. If this address is allocated through DHCP, the host should support DHCPv4 over tunnel. After the allocation, the host periodically sends pinhole packet to the concentrator to install the address mapping and keep it alive.

For IPv4 data traffic, the host performs the IPv4-in-IPv6 encapsulation and decapsulation on the tunnel interface. When sending out an IPv4 packet, it performs the encapsulation, using the IPv6 address of the 4over6 concentrator as the IPv6 destination address, and its own IPv6 address as the IPv6 source address. The encapsulated packet will be forwarded to the IPv6 network. The decapsulation on 4over6 initiator is simple. When receiving an IPv4-in-IPv6 packet, the initiator just drops the IPv6 header, and hands it to upper layer.

5.2.2. CPE initiator

The CPE case is quite similar to the host initiator case. The CPE assign the allocated IPv4 address to its tunnel interface. The local IPv4 network won't take part in the public IPv4 allocation; instead, end hosts will use private IPv4 addresses, possibly allocated by the CPE. After the allocation, the CPE periodically sends pinhole packet to the concentrator to install the address mapping and keep it alive.

On data plan, the CPE can be viewed as a regular IPv4 NAT(using tunnel interface as the NAT outside interface) cascaded with a tunnel initiator. For IPv4 data packets received from the local network, the CPE translates these packets, using the tunnel interface address as the source address, and then encapsulates the translated packet into IPv6, using the concentrator's IPv6 address as the destination address, the CPE's IPv6 address as source address. For IPv6 data packet received from the IPv6 network, the CPE performs decapsulation and IPv4 public-to-private translation. As to the CPE itself, it uses the public, tunnel interface address to communicate with the

IPv4 Internet, and the private, IPv4 interface address to communicate with the local network.

5.3. 4over6 concentrator behavior

4over6 concentrator represents the IPv4-IPv6 border router working as the remote tunnel endpoint for 4over6 initiators, with its IPv6 interface connected to the IPv6 network, IPv4 interface connected to the IPv4 Internet, and a tunnel interface supporting IPv4-in-IPv6 encapsulation and decapsulation. There’s no CGN on the 4over6 concentrator, it won’t perform any translation function; instead, 4over6 concentrator maintains an IPv4-IPv6 address mapping table for IPv4 data encapsulation.

4over6 concentrator maintains the address mapping according to the initiators’ demand. When receiving a pinhole packet from an initiator, the concentrator reads the IPv4 and IPv6 source addresses from the packet, install the mapping entry into the mapping table or renew it if it already exists. When the lifetime of a mapping entry expires, the concentrator deletes it from the table. So the initiator should send pinhole packet with an interval shorter than the lifetime of the mapping entry. The mapping entry is used to provide correct encapsulation destination address for concentrator encapsulation. As long as the entry exists in the table, the concentrator can encapsulate inbound IPv4 packets destined to the initiator, with the initiator’s IPv6 address as IPv6 destination.

On the IPv6 side, 4over6 concentrator decapsulates IPv4-in-IPv6 packets coming from 4over6 initiators. It removes the IPv6 header of every IPv4-in-IPv6 packet and forwards it to the IPv4 Internet. On the IPv4 side, the concentrator encapsulates the IPv4 packets destined to 4over6 initiators. When performing the IPv4-in-IPv6 encapsulation, the concentrator uses its own IPv6 address as the IPv6 source address, uses the IPv4 destination address in the packet to look up IPv6 destination address in the address mapping table. After the encapsulation, the concentrator sends the IPv6 packet on its IPv6 interface to reach an initiator.

The 4over6 concentrator, or its upstream router should advertise the IPv4 prefix which contains the IPv4 addresses of 4over6 users to the IPv4 side, in order to make these initiators reachable on IPv4 Internet.

Since the concentrator has to maintain the IPv4-IPv6 address mapping table, the concentrator is stateful in IP level. Note that this table will be much smaller than a CGN table, as there is no port information involved.

6. Technical advantages

Public 4over6 provides a method for users in IPv6 network to communicate with IPv4. In many scenarios, this can be viewed as an alternative to IPv6-IPv4 translation mechanisms which have well-known limitations described in [RFC4966] .

Since a 4over6 initiator uses a public IPv4 address, Public 4over6 supports full bidirectional communication between IPv4 Internet and hosts/IPv4 networks in IPv6 access network. In particular, it supports the servers in IPv6 network to provide IPv4 application service transparently.

Public 4over6 provides IPv4 access over IPv6 network while keeps IPv4-IPv6 addressing and routing separated. Therefore the ISP can manage the native IPv6 network independently without the influence of IPv4-over-IPv6 requirements, and also provision IPv4 in a flexible, on-demand way.

Public 4over6 supports dynamic reuse of a single IPv4 address between multiple subscribers based on their dynamic requirement of communicating with IPv4 Internet. A subscriber will request a public IPv4 address for a period of time only when it need to communicate with IPv4 Internet. Besides, in the CPE case, one public IPv4 address will be shared by the local network. So Public 4over6 can improve the reuse rate of IPv4 addresses.

Public 4over6 is suited for network users/ISPs which can still get/provide public IPv4 addresses. Dual-stack lite is suited for network users/ISPs which can no longer get/provide public IPv4 addresses. By combining Public 4over6 and Dual-stack lite, the IPv4-over-IPv6 Hub and spoke problem can be well solved.

7. Acknowledgement

The authors would like to thank Alain Durand and Dan Wing for their valuable comments on this draft.

8. References

8.1. Normative References

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

[RFC4925] Li, X., Dawkins, S., Ward, D., and A. Durand, "Softwire Problem Statement", RFC 4925, July 2007.

[RFC4966] Aoun, C. and E. Davies, "Reasons to Move the Network Address Translator - Protocol Translator (NAT-PT) to Historic Status", RFC 4966, July 2007.

[RFC5549] Le Faucheur, F. and E. Rosen, "Advertising IPv4 Network Layer Reachability Information with an IPv6 Next Hop", RFC 5549, May 2009.

[RFC5565] Wu, J., Cui, Y., Metz, C., and E. Rosen, "Softwire Mesh Framework", RFC 5565, June 2009.

8.2. Informative References

[I-D.cui-softwire-b4-translated-ds-lite]
Cui, Y., Wu, J., and D. Wu, "B4 translated DS-lite enable AFTR to serve more B4s", draft-cui-softwire-b4-translated-ds-lite-00 (work in progress), October 2010.

[I-D.cui-softwire-dhcp-over-tunnel]
Cui, Y., Wu, P., and J. Wu, "DHCPv4 Behavior over IP-IP tunnel", draft-cui-softwire-dhcp-over-tunnel-00 (work in progress), June 2011.

[I-D.ietf-pcp-base]
Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", draft-ietf-pcp-base-13 (work in progress), July 2011.

[I-D.ietf-softwire-ds-lite-tunnel-option]
Hankins, D. and T. Mrugalski, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Option for Dual- Stack Lite", draft-ietf-softwire-ds-lite-tunnel-option-10 (work in progress), March 2011.

[I-D.ietf-softwire-dual-stack-lite]
Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4

Exhaustion", draft-ietf-softwire-dual-stack-lite-11 (work in progress), May 2011.

[I-D.sun-v6ops-laft6]
Sun, Q. and C. Xie, "LAFT6: Lightweight address family transition for IPv6", draft-sun-v6ops-laft6-01 (work in progress), March 2011.

Authors' Addresses

Yong Cui
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R.China

Phone: +86-10-6260-3059
Email: yong@csnet1.cs.tsinghua.edu.cn

Jianping Wu
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R.China

Phone: +86-10-6278-5983
Email: jianping@cernet.edu.cn

Peng Wu
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R.China

Phone: +86-10-6278-5822
Email: weapon@csnet1.cs.tsinghua.edu.cn

Chris Metz
Cisco Systems, Inc.
3700 Cisco Way
San Jose, CA 95134
USA

Email: chmetz@cisco.com

Olivier Vautrin
Juniper Networks
1194 N Mathilda Avenue
Sunnyvale, CA 94089
USA

Email: Olivier@juniper.net

Yiu L. Lee
Comcast
One Comcast Center
Philadelphia, PA 19103
USA

Email: yiu_lee@cable.comcast.com