    4over6 Transit Solution Using IP Encapsulation and MP-BGP Extensions

Abstract

   The emerging and growing deployment of IPv6 networks will introduce
   cases where connectivity with IPv4 networks crossing IPv6 transit
   backbones is desired. This document describes a mechanism for
   automatic discovery and creation of IPv4-over-IPv6 tunnels via
   extensions to multiprotocol BGP. It is targeted at connecting
   islands of IPv4 networks across an IPv6-only backbone without the
   need for a manually configured overlay of tunnels. The mechanisms
   described in this document have been implemented, tested, and
   deployed on the large research IPv6 network in China.

Status of This Memo

IESG Note

   The mechanisms and techniques described in this document are related
   to specifications developed by the IETF softwire working group and
   published as Standards Track documents by the IETF, but the
   relationship does not prevent publication of this document.

Table of Contents

1. Introduction

   Due to the lack of IPv4 address space, more and more IPv6 networks
   have been deployed not only on edge networks but also on backbone
   networks. However, there are still a large number of legacy IPv4
   hosts and applications. As a result, IPv6 networks and IPv4
   applications/hosts will have to coexist for a long period of time.

   The emerging and growing deployment of IPv6 networks will introduce
   cases where connectivity with IPv4 networks is desired. Some IPv6
   backbones will need to offer transit services to attached IPv4 access
   networks. The method to achieve this would be to encapsulate and
   then transport the IPv4 payloads inside IPv6 tunnels spanning the
   backbone. There are some IPv6/IPv4-related tunneling protocols and
   mechanisms defined in the literature. But at the time that the
   mechanism described in this document was introduced, most of these
   existing techniques focused on the problem of IPv6 over IPv4, rather
   than the case of IPv4 over IPv6. Encapsulation methods alone, such
   as those defined in [RFC2473], require manual configuration in order
   to operate. When a large number of tunnels are necessary, manual
   configuration can become burdensome. To the above problem, this
   document describes an approach, referred to as "4over6".

   The 4over6 mechanism concerns two aspects: the control plane and the
   data plane. The control plane needs to address the problem of how to
   set up an IPv4-over-IPv6 tunnel in an automatic and scalable fashion
   between a large number of edge routers. This document describes
   experimental extensions to Multiprotocol Extension for BGP (MP-BGP)
   [RFC4271] [RFC4760] employed to communicate tunnel endpoint
   information and establish 4over6 tunnels between dual-stack Provider
   Edge (PE) routers positioned at the edge of the IPv6 backbone
   network. Once the 4over6 tunnel is in place, the data plane focuses
   on the packet forwarding processes of encapsulation and
   decapsulation.

2. 4over6 Framework Overview

   In the topology shown in Figure 1, a number of IPv6-only P routers
   compose a native IPv6 backbone. The PE routers are dual stack and
   referred to as 4over6 PE routers. The IPv6 backbone acts as a
   transit core to transport IPv4 packets across the IPv6 backbone.
   This enables each of the IPv4 access islands to communicate with one
   another via 4over6 tunnels spanning the IPv6 transit core.

```
                _._._._._                  _._._._._
               | IPv4    |                | IPv4    |
               | access  |                | access  |
               | island  |                | island  |
                _._._._._                  _._._._._
                    |                          |
               Dual-Stack                 Dual-Stack
               "4over6 PE"                "4over6 PE"
                    |                          |
                    |                          |
                __+_____+__
       4over6   /  :   :              :   :  \    IPv6 only
       Tunnels | :        :        :        : | transit core
       between | :            [P]        :    | with multiple
          PEs  | :        :        :        : |  [P routers]
               | :  :          :      :   :   |
                \_._._._._._._._._._._._._._./
                   | | /              \ |
                   |                    |
               Dual-Stack          Dual-Stack
               "4over6 PE"         "4over6 PE"
                 | |                   |
                _._._._._            _._._._._
               | IPv4    |          | IPv4    |
               | access  |          | access  |
               | island  |          | island  |
                _._._._._            _._._._._
```
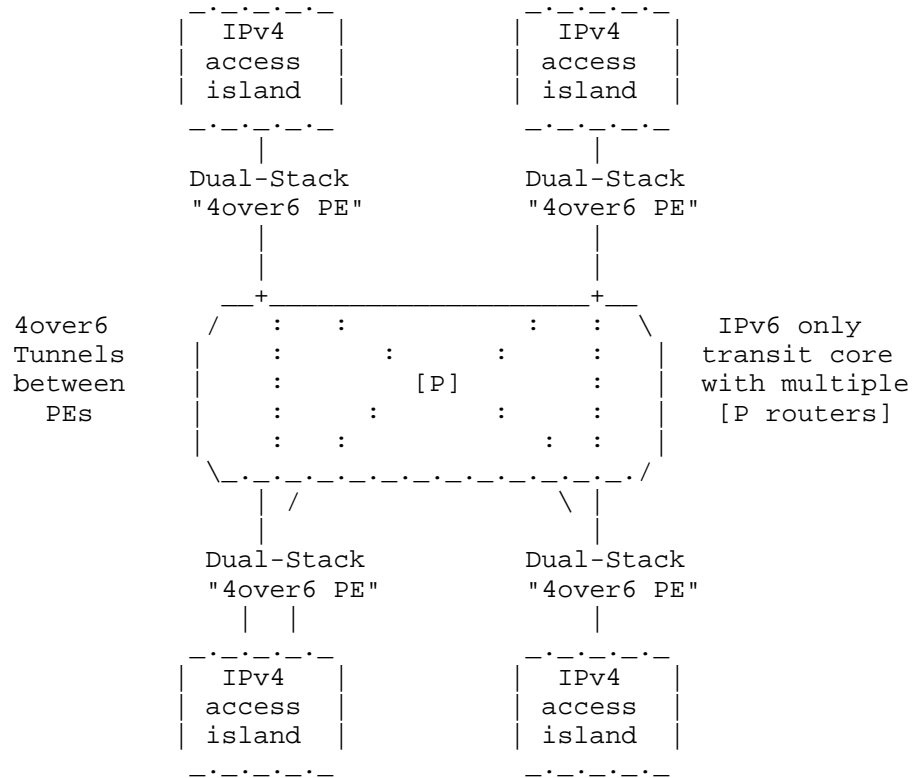
                  Figure 1: IPv4 over IPv6 Network Topology

   As shown in Figure 1, there are multiple dual-stack PE routers
   connected to the IPv6 transit core. In order for the ingress 4over6
   PE router to forward an IPv4 packet across the IPv6 backbone to the
   correct egress 4over6 PE router, the ingress 4over6 PE router must
   learn which IPv4 destination prefixes are reachable through each
   egress 4over6 PE router. MP-BGP will be extended to distribute the
   destination IPv4 prefix information between peering dual-stack PE
   routers. Section 4 of this document presents the definition of the
   4over6 protocol field in MP-BGP, and Section 5 describes MP-BGP's
   extended behavior in support of this capability.

   After the ingress 4over6 PE router learns the correct egress 4over6
   PE router via MP-BGP, it will forward the packet across the IPv6
   backbone using IP encapsulation. The egress 4over6 PE router will
   receive the encapsulated packet, remove the IPv6 header, and then
   forward the original IPv4 packet to its final IPv4 destination.
   Section 6 describes the procedure of packet forwarding.

3. Prototype Implementation

   An implementation of the 4over6 mechanisms described in this document
   was developed, tested, and deployed on Linux with kernel version 2.4.
   The prototype system is composed of three components: packet
   forwarding, the encapsulation table, and MP-BGP extensions. The
   packet forwarding and encapsulation table are Linux kernel modules,
   and the MP-BGP extension was developed by extending Zebra routing
   software.

   The following sections will discuss these parts in detail.

3.1. 4over6 Packet Forwarding

   Forwarding an IPv4 packet through the IPv6 transit core includes
   three parts: encapsulation of the incoming IPv4 packet with the IPv6
   tunnel header, transmission of the encapsulated packet over the IPv6
   transit backbone, and decapsulation of the IPv6 header and forwarding
   of the original IPv4 packet. Native IPv6 routing and forwarding are
   employed in the backbone network since the P routers take the 4over6
   tunneled packets as just native IPv6 packets. Therefore, 4over6
   packet forwarding involves only the encapsulation process and the
   decapsulation process, both of which are performed on the 4over6 PE
   routers.

```
                Tunnel from Ingress PE to Egress PE
                  ---------------------------->
                Tunnel                         Tunnel
                Entry-Point                    Exit-Point
                Node                           Node
  +-+    IPv4    +--+   IPv6 Transit Core   +--+    IPv4    +-+
  |S|-->--//-->--|PE|=====>=====//=====>=====|PE|-->--//-->--|D|
  +-+            +--+                       +--+            +-+
  Original    Ingress PE                    Egress PE       Original
  Packet     (Encapsulation)              (Decapsulation)   Packet
  Source                                                    Destination
  Node                                                      Node
```
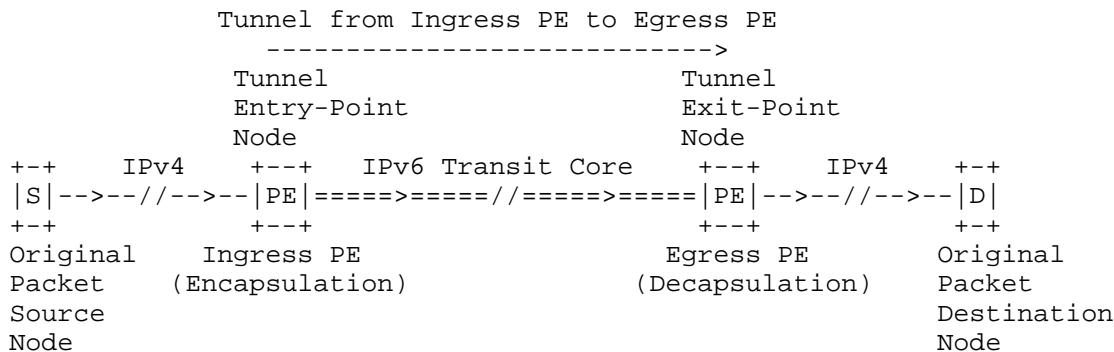
              Figure 2: Packet Forwarding along 4over6 Tunnel

   As shown in Figure 2, packet encapsulation and decapsulation are both
   on the dual-stack 4over6 PE routers. Figure 3 shows the format of
   packet encapsulation and decapsulation.

```
                           +-----------------------------------//-----+
                           | IPv4 Header |   Packet Payload           |
                           +-----------------------------------//-----+
                            <              Original IPv4 Packet       >
                                      |
                                      |(Encapsulation on ingress PE)
                                      |
                                      v
   < Tunnel IPv6 Headers > <            Original IPv4 Packet          >
   +-----------+ - - - - - +------------+-----------//--------------+
   |   IPv6    | IPv6      |   IPv4     |                           |
   |           | Extension |            |        Packet Payload     |
   |  Header   | Headers   |  Header    |                           |
   +-----------+ - - - - - +------------+-----------//--------------+
    <                      Tunnel IPv6 Packet                      >
                                      |
                                      |(Decapsulation on egress PE)
                                      |
                                      v
                           +-----------------------------------//-----+
                           | IPv4 Header |   Packet Payload           |
                           +-----------------------------------//-----+
                            <              Original IPv4 Packet       >
```
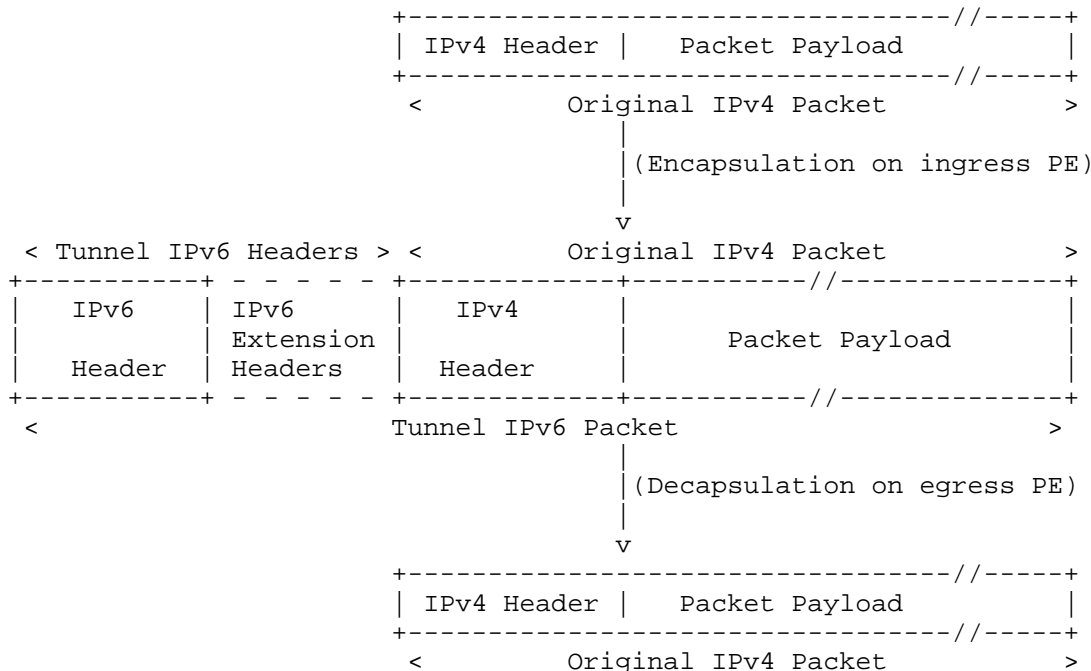
   Figure 3: Packet Encapsulation and Decapsulation on Dual-Stack 4over6
             PE Router

   The encapsulation format to apply is IPv4 encapsulated in IPv6, as
   outlined in [RFC2473].

3.2. Encapsulation Table

   Each 4over6 PE router maintains an encapsulation table as depicted in
   Figure 4. Each entry in the encapsulation table consists of an IPv4
   prefix and its corresponding IPv6 address. The IPv4 prefix is a
   particular network located in an IPv4 access island network. The
   IPv6 address is the 4over6 virtual interface (VIF) address of the
   4over6 PE router that the IPv4 prefix is reachable through. The
   encapsulation table is built and maintained using local configuration
   information and MP-BGP advertisements received from remote 4over6 PE
   routers.

   The 4over6 VIF is an IPv6 /128 address that is locally configured on
   each 4over6 router. This address, as an ordinary global IPv6
   address, must also be injected into the IPv6 IGP so that it is
   reachable across the IPv6 backbone.

```
+-------------+-----------------------------------------------+
| IPv4 Prefix | IPv6 Advertising Address Family Border Router |
+-------------+-----------------------------------------------+
```

Figure 4: Encapsulation Table

When an IPv4 packet arrives at the ingress 4over6 PE router, a lookup
in the local IPv4 routing table will result in a pointer to the local
encapsulation table entry with the matching destination IPv4 prefix.
There is a corresponding IPv6 address in the encapsulation table.
The IPv4 packet is encapsulated in an IPv6 header. The source
address in the IPv6 header is the IPv6 VIF address of the local
4over6 PE router and the destination address is the IPv6 VIF address
of the remote 4over6 PE router contained in the local encapsulation
table. The packet is then subjected to normal IPv6 forwarding for
transport across the IPv6 backbone.

When the encapsulated packet arrives at the egress 4over6 PE router,
the IPv6 header is removed and the original IPv4 packet is forwarded
to the destination IPv4 network based on the outcome of the lookup in
the IPv4 routing table contained in the egress 4over6 PE router.

3.3. MP-BGP 4over6 Protocol Extensions

Each 4over6 PE router possesses an IPv4 interface connected to an
IPv4 access network(s). It can peer with other IPv4 routers using
IGP or BGP routing protocols to exchange local IPv4 routing
information. Routing information can also be installed on the 4over6
PE router using static configuration methods.

Each 4over6 PE also possesses at least one IPv6 interface to connect
it into the IPv6 transit backbone. The 4over6 PE typically uses IGP
routing protocols to exchange IPv6 backbone routing information with
other IPv6 P routers. The 4over6 PE router will also form an MP-iBGP
(Internal BGP) peering relationship with other 4over6 PE routers
connected to the IPv6 backbone network.

The use of MP-iBGP suggests that the participating 4over6 PE routers
that share a route reflector or form a full mesh of TCP connections
are contained in the same autonomous system (AS). This
implementation is in fact only deployed over a single AS. This was
not an intentional design constraint but rather reflected the single
AS topology of the CNGI-CERNET2 (China Next Generation Internet -
China Education and Research Network) national IPv6 backbone used in
the testing and deployment of this solution.

3.3.1. Receiving Routing Information from Local CE

   When a 4over6 PE router learns routing information from the locally
   attached IPv4 access networks, the 4over6 MP-iBGP entity should
   process the information as follows:

   1. Install and maintain local IPv4 routing information in the IPv4
       routing database.

   2. Install and maintain new entries in the encapsulation table.
       Each entry should consist of the IPv4 prefix and the local IPv6
       VIF address.

   3. Advertise the new contents of the local encapsulation table in
       the form of MP_REACH_NLRI update information to remote 4over6 PE
       routers. The format of these updates is as follows:

       * AFI = 1 (IPv4)

       * SAFI = 67 (4over6)

       * NLRI = IPv4 network prefix

       * Network Address of Next Hop = IPv6 address of its 4over6 VIF

   4. A new Subsequent Address Family Identifier (SAFI) BGP 4over6 (67)
       has been assigned by IANA. We call a BGP update with a SAFI of
       67 as 4over6 routing information.

3.3.2. Receiving 4over6 Routing Information from a Remote 4over6 PE

   A local 4over6 PE router will receive MP_REACH_NLRI updates from
   remote 4over6 routers and use that information to populate the local
   encapsulation table and the BGP routing database. After validating
   the correctness of the received attribute, the following procedures
   are used to update the local encapsulation table and redistribute new
   information to the local IPv4 routing table:

   1. Validate the received BGP update packet as 4over6 routing
       information by AFI = 1 (IPv4) and SAFI = 67 (4over6).

   2. Extract the IPv4 network address from the NLRI field and install
       as the IPv4 network prefix.

   3. Extract the IPv6 address from the Network Address of the Next Hop
       field and place that as an associated entry next to the IPv4
       network index. (Note, this describes the update of the local
       encapsulation table.)

4. Install and maintain a new entry in the encapsulation table with
    the extracted IPv4 prefix and its corresponding IPv6 address.

5. Redistribute the new 4over6 routing information to the local IPv4
    routing table. Set the destination network prefix as the
    extracted IPv4 prefix, set the Next Hop as Null, and Set the
    OUTPUT Interface as the 4over6 VIF on the local 4over6 PE router.

Therefore, when an ingress 4over6 PE router receives an IPv4 packet,
the lookup in its IPv4 routing table will have a result of the output
interface as the local 4over6 VIF, where the incoming IPv4 packet
will be encapsulated with a new IPv6 header, as indicated in the
encapsulation table.

4. 4over6 Deployment Experience

4.1. CNGI-CERNET2

A prototype of the 4over6 solution is implemented and deployed on
CNGI-CERNET2. CNGI-CERNET2 is one of the China Next Generation
Internet (CNGI) backbones, operated by the China Education and
Research Network (CERNET). CNGI-CERNET2 connects approximately 25
core nodes distributed in 20 cities in China at speeds of 2.5-10
Gb/s. The CNGI-CERNET2 backbone is IPv6-only with some attached
customer premise networks (CPNs) being dual stack. The CNGI-CERNET2
backbone, attached CNGI-CERNET2 CPNs, and CNGI-6IX Exchange all have
globally unique AS numbers. This IPv6 backbone is used to provide
transit IPv4 services for customer IPv4 networks connected via 4over6
PE routers to the backbone.

4.2. 4over6 Testbed on the CNGI-CERNET2 IPv6 Network

Figure 5 shows 4over6 deployment network topology.

```
      +-------------------------------------------------------+
      |                    IPv6 (CERNET2)                     |
      |                                                       |
      +-------------------------------------------------------+
          |               |                 |             |
    Tsinghua|Univ.     Peking|Univ.        SJTU|      Southeast|Univ.
      +------+            +------+            +------+          +------+
      |4over6|    ...     |4over6|            |4over6|   ...    |4over6|
      |router|            |router|            |router|          |router|
      +------+            +------+            +------+          +------+
          |                  |                  |                |
          |                  |                  |                |
          |                  |                  |                |
    +-----------+      +-----------+      +-----------+    +-----------+
    |IPv4 access|  ... |IPv4 access|      |IPv4 access| ...|IPv4 access|
    |  network  |      |  network  |      |  network  |    |  network  |
    +-----------+      +-----------+      +-----------+    +-----------+
          |
    +---------------------+
    |    IPv4 (Internet)  |
    |                     |
    +---------------------+
```
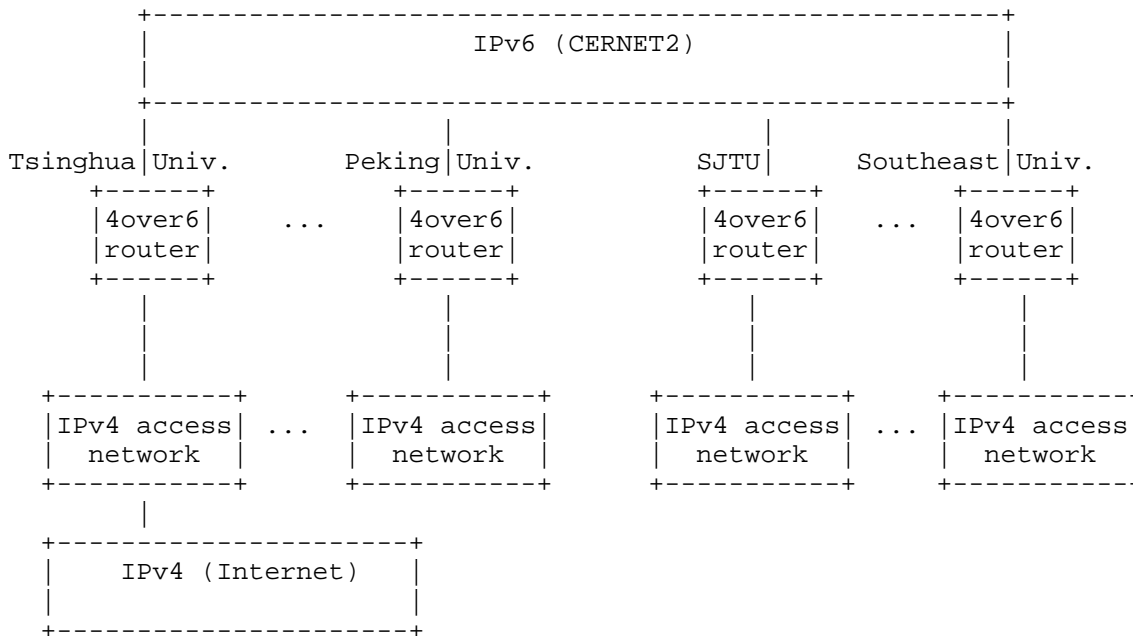
Figure 5: 4over6 Deployment Network Topology

The IPv4-only access networks are equipped with servers and clients
running different applications. The 4over6 PE routers are deployed
at 8 x IPv6 nodes of CNGI-CERNET2, located in seven universities and
five cities across China. As suggested in Figure 5, some of the IPv4
access networks are connected to both IPv6 and IPv4 networks, and
others are only connected to the IPv6 backbone. In the deployment,
users in different IPv4 networks can communicate with each other
through 4over6 tunnels.

4.3. Deployment Experiences

A number of 4over6 PE routers were deployed and configured to support
the 4over6 transit solution. MP-BGP peerings were established, and
successful distribution of 4over6 SAFI information occurred.
Inspection of the BGP routing and encapsulation tables confirmed that
the correct entries were sent and received. ICMP ping traffic
indicated that IPv4 packets were successfully transiting the IPv6
backbone.

In addition, other application protocols were successfully tested per
the following:

   o HTTP. A client running Internet Explorer in one IPv4 client
      network was able to access and download multiple objects from an
      HTTP server located in another IPv4 client network.

   o P2P. BitComet software running on several PCs placed in different
      IPv4 client networks were able to find each other and share files.

   Other protocols, including FTP, SSH, IM (e.g., MSN, Google Talk), and
   Multimedia Streaming, all functioned correctly.

5. Ongoing Experiment

   Based on the above successful experiment, we are going to have
   further experiments in the following two aspects.

   1. Inter-AS 4over6

       The above experiment is only deployed over a single AS. With the
       growth of the network, there could be multiple ASes between the
       edge networks. Specifically, the Next Hop field in MP-BGP
       indicates the tunnel endpoint in the current 4over6 technology.
       However, in the Inter-AS scenario, the tunnel endpoint needs to be
       separated from the field of Next Hop. Moreover, since the
       technology of 4over6 is deployed on the router running MP-BGP, the
       supportability of 4over6 on each Autonomous System Border Router
       (ASBR) will be a main concern in the Inter-AS experiment. We may
       consider different situations: (1) Some ASBRs do not support
       4over6; (2) ASBRs only support the 4over6 control plane (i.e., MP-
       BGP extension of 4over6) rather than 4over6 data plane; (3) ASBRs
       support both the control plane and the data plane for 4over6.

   2. Multicast 4over6

       The current 4over6 technology only supports unicast routing and
       data forwarding. With the deployment of network-layer multicast
       in multiple IPv4 edge networks, we need to extend the 4over6
       technology to support multicast including both multicast tree
       manipulation on the control plane and multicast traffic forwarding
       on the data plane. Based on the current unicast 4over6 technology
       providing the unicast connectivity of edge networks over the
       backbone in another address family, the multicast 4over6 will
       focus on the mapping technologies between the multicast groups in
       the different address families.

6. Relationship to Softwire Mesh Effort

   The 4over6 solution was presented at the IETF Softwires Working Group
   Interim meeting in Hong Kong in January 2006. The existence of this
   large-scale implementation and deployment clearly showed that MP-BGP
   could be employed to support tunnel setup in a scalable fashion
   across an IPv6 backbone. Perhaps most important was the use-case
   presented -- an IPv6 backbone that offers transit to attached client
   IPv4 networks.

   The 4over6 solution can be viewed as a precursor to the Softwire Mesh
   Framework proposed in the softwire problem statement [RFC4925].
   However, there are several differences with this solution and the
   effort that emerged from the Softwires Working Group called "softwire
   Mesh Framework" [RFC5565] and the related solutions [RFC5512]
   [RFC5549].

   o MP-BGP Extensions. 4over6 employs a new SAFI (BGP 4over6) to
      convey client IPv4 prefixes between 4over6 PE routers. Softwire
      Mesh retains the original AFI-SAFI designations, but it uses a
      modified MP_REACH_NLRI format to convey IPv4 Network Layer
      Reachability Information (NLRI) prefix information with an IPv6
      next_hop address [RFC5549].

   o Encapsulation. 4over6 assumes IP-in-IP or it is possible to
      configure Generic Routing Encapsulation (GRE). Softwires uses
      those two scenarios configured locally or for IP headers that
      require dynamic updating. As a result, the BGP encapsulation SAFI
      is introduced in [RFC5512].

   o Multicast. The basic 4over6 solution only implemented unicast
      communications. The multicast communications are specified in the
      Softwire Mesh Framework and are also supported by the multicast
      extension of 4over6.

   o Use-Cases. The 4over6 solution in this document specifies the
      4over6 use-case, which is also pretty easy to extend for the use-
      case of 6over4. The Softwire Mesh Framework supports both 4over6
      and 6over4.

7. IANA Considerations

   A new SAFI value (67) has been assigned by IANA for the BGP 4over6
   SAFI.

8. Security Considerations

   Tunneling mechanisms, especially automatic ones, often have potential
   problems of Distributed Denial of Service (DDoS) attacks on the
   tunnel entry-point or tunnel exit-point. As the advantage, the BGP
   4over6 extension doesn't allocate resources to a single flow or
   maintain the state for a flow. However, since the IPv6 tunnel
   endpoints are globally reachable IPv6 addresses, it would be trivial
   to spoof IPv4 packets by encapsulating and sending them over IPv6 to
   the tunnel interface. This could bypass IPv4 Reverse Path Forwarding
   (RPF) or other antispoofing techniques. Also, any IPv4 filters may
   be bypassed.

   An iBGP peering relationship may be maintained over IPsec or other
   secure communications.

9. Conclusion

   The emerging and growing deployment of IPv6 networks, in particular,
   IPv6 backbone networks, will introduce cases where connectivity with
   IPv4 networks is desired. Some IPv6 backbones will need to offer
   transit services to attached IPv4 access networks. The 4over6
   solution outlined in this document supports such a capability through
   an extension to MP-BGP to convey IPv4 routing information along with
   an associated IPv6 address. Basic IP encapsulation is used in the
   data plane as IPv4 packets are tunneled through the IPv6 backbone.

   An actual implementation has been developed and deployed on the CNGI-
   CERNET2 IPv6 backbone.

10. Acknowledgements

   During the design procedure of the 4over6 framework and definition of
   BGP-MP 4over6 extension, Professor Ke Xu gave the authors many
   valuable comments. The support of the IETF Softwires WG is also
   gratefully acknowledged with special thanks to David Ward, Alain
   Durand, and Mark Townsley for their rich experience and knowledge in
   this field. Yakov Rekhter provided helpful comments and advice.
   Mark Townsley reviewed this document carefully and gave the authors a
   lot of valuable comments, which were very important for improving
   this document.

   The deployment and test for the prototype system was conducted among
   seven universities -- namely, Tsinghua University, Peking University,
   Beijing University of Post and Telecommunications, Shanghai Jiaotong
   University, Huazhong University of Science and Technology, Southeast

University, and South China University of Technology. The authors
would like to thank everyone involved in this effort at these
universities.

11. Normative References

   [RFC2473]   Conta, A. and S. Deering, "Generic Packet Tunneling in
               IPv6 Specification", RFC 2473, December 1998.

   [RFC4271]   Rekhter, Y., Li, T., and S. Hares, "A Border Gateway
               Protocol 4 (BGP-4)", RFC 4271, January 2006.

   [RFC4760]   Bates, T., Chandra, R., Katz, D., and Y. Rekhter,
               "Multiprotocol Extensions for BGP-4", RFC 4760,
               January 2007.

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

   [RFC5512]   Mohapatra, P. and E. Rosen, "The BGP Encapsulation
               Subsequent Address Family Identifier (SAFI) and the BGP
               Tunnel Encapsulation Attribute", RFC 5512, April 2009.

   [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.

Authors' Addresses

    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

    Yong Cui
    Tsinghua University
    Department of Computer Science, Tsinghua University
    Beijing 100084
    P.R. China
    Phone: +86-10-6278-5822
    EMail: cy@csnet1.cs.tsinghua.edu.cn

    Xing Li
    Tsinghua University
    Department of Electronic Engineering, Tsinghua University
    Beijing 100084
    P.R. China
    Phone: +86-10-6278-5983
    EMail: xing@cernet.edu.cn

    Mingwei Xu
    Tsinghua University
    Department of Computer Science, Tsinghua University
    Beijing 100084
    P.R. China
    Phone: +86-10-6278-5822
    EMail: xmw@csnet1.cs.tsinghua.edu.cn

    Chris Metz
    Cisco Systems, Inc.
    3700 Cisco Way
    San Jose, CA 95134
    USA
    EMail: chmetz@cisco.com