

IPv6 - The Road Ahead

Executive Summary

IPv6 is the next step in the evolution of the internet, which currently runs on IPv4 (IPv5 was never implemented), one which many carriers are increasingly eager to take for both practical, technical reasons as well as political. One of the biggest issues addressed by IPv6 is the inadequate number of IP addresses provided by IPv4, an issue which is particularly acute in Asia. Riverstone has a 3 Phase Plan to introduce IPv6 functionality, starting with basic connectivity and expanding to include comprehensive IPv6 support including multicast, MPLS, OSPF and BGP support.

Introduction

IPv6 is gaining momentum worldwide, driven by the need for more IP addresses than IPv4 can provide. This is particularly true in Asia where shortage of IPv4 address is more evident and the strategic importance of IPv6 is promoted by the governments. Indeed, even the US Government is getting on board with IPv6, with the DoD mandating upgrade by 2008 and other government organs likely to do the same [Network World]. By the end of 2007, China is expected to top the world with 57 million broadband subscribers [CNet News]. However, China has fewer public IPv4 addresses than the universities in Michigan (i.e. MichNet). Network Address Translation (NAT) has been a workable but imperfect solution for this IP address shortage for a number of years. However, the proliferation of mobile communication devices, the increasing popularity of peer-to-peer networking, online gaming, and the emerging integration of home appliances into a ubiquitous network in combination with the often significant limitations of NAT, are creating real demands for IPv6. IPv6 will allow for 340,282,366,920,938,463,463,374,607,431,768,211,456 total theoretically assignable addresses, which implies that approximately 3,700,000,000,000,000,000,000,000 IP addresses can be assigned per square inch of the earth's surface. Therefore IPv6 should resolve the IP address shortage issue once and for all.

The adoption of IPv6 creates an opportunity for Riverstone Networks to provide additional solutions for existing customers as well as opening the doors to new customers as well. This document describes Riverstone's 3 phase plan to meet customer demand for IPv6 support. It emphasizes providing a complete IPv6 solution rather than just giving SPs bits and pieces and letting them figure out what/how to do with the bits and pieces.

Worldwide IPv6 Deployment Status

Many research & education networks worldwide have been conducting IPv6 trials for years. And they have started moving to the next phase of

IPv6 – Riverstone's Phased Approach

Phase 1

Currently, the 15008 fully supports IPv6 data plane functionalities. On the control plane, ROS-X supports static routing.

Phase 2

In Release 2.0, ROS-X provides the fundamental IPv6 dynamic routing and network management functionalities. The goal of phase-2 is to enable customers to conduct proof of concept or labs trials of basic commercial IPv6 services. The major features supported in 2.0 include:

- DNS for IPv6
- DHCP v6
- Dual protocol stacks (IPv4 and IPv6)
- Neighbor Discovery
- Dynamic routing: i/IS-IS, MP-BGP, policy based routing
- ACLs for IPv6
- MPLS tunneling of IPv6
- Management: SNMP/MIBs, ICMP v6, IPv6 path MTU discovery

Phase 3

After Release 2.0, ROS-X will provide a comprehensive IPv6 solution. The goal of phase-3 is to enable customers to deploy sophisticated commercial IPv6 services. The major additional features we plan to support include:

- VRRP v3
- IPv6 over IPv4 tunneling
- IPv6 over IPv4 GRE tunneling
- 6to4 tunneling
- OSPF v3
- IPv6 QoS
- IPv6 multicast
- Stateless Address Auto-configuration
- IPv6 over POS links
- BGP-MPLS VPN extension for IPv6 VPNs
- IPv4-compatible tunneling

Mobile devices and newly networked consumer electronics are helping drive the demand for more and more unique IP addresses, a demand which due to the limited IPv4 address space is helping drive the demand for IPv6 services

IPv6 deployment - upgrading all their network devices to support native IPv6 routing and forwarding. The well-known IPv6 projects include Internet2/Abilene and 6Bone in the U.S., 6Net and Euro6IX in Europe, WIDE in Japan, KOREN IPv6 in Korea, CERNet in China, etc.

Commercially, many manufacturers of consumer electronics such as Sony, NEC, Samsung are very enthusiastic about supporting IPv6 in their new products. So are many consumer software vendors such as Microsoft, Sun, and Linux publishers. Some Service Providers in Asia such as NTT and KDDI have also started some IPv6 testing and offering preliminary IPv6 hosting and gateway services. The following chart gives an overview of the current IPv6 deployment status in major economies worldwide:

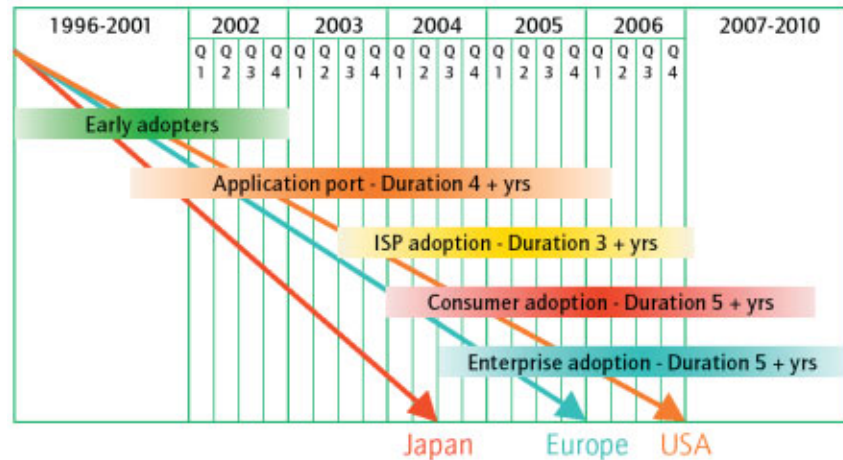


Figure 1: IPv6 Adoption. Source, IPv6 Cluster

However, there are still 2 factors that are hampering major service providers from offering commercial IPv6 services:

1. The lack of commercial applications requiring IPv6. The inter-dependency between commercial IPv6 applications and IPv6-capable network infrastructure is a chicken-and-egg dilemma.
2. The complexity and cost associated with the transition from IPv4 to IPv6. The complexity arises from potential stability and security issues during the transition period, lack of IPv6 network management solutions and operational experience, etc. The cost arises from additional functionalities and equipment, employee training, etc.

For these 2 reasons, successful commercial IPv6 services are still in their infancy.

However, the situation is changing. For the first factor, mobile communication (WiFi, WiMax, 3G), online gaming, IP TV and home networking have started to gain strong momentum since the second half of 2004. These applications cannot reach their full potential with IPv4 and NAT. Therefore they may become the IPv6 killer applications, and lead to the breaking of the dilemma. Once the dilemma is removed, because IPv6 is anticipated by so many people for so long, the flood-gate could be opened from that point. The second factor becomes less of an issue when there is demand for IPv6 services. Plus, IPv6 can simplify IP routing (e.g. better IP address summarization, elimination of NAT) and lead to lower OpEx eventually, according to the following figure. Therefore, it is possible that the IPv6 inflection point is getting close.

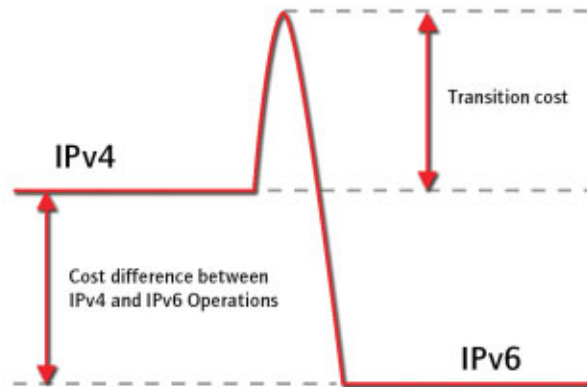


Figure 2: Cost of Transitioning from IPv4 to IPv6 Services

IPv6 Technical Overview

This section describes the IPv6 header and address format.

IPv6 Header

An IPv6 header is shown below, where:

- “Flow Label” is a 20-bit field which can be used to identify a traffic flow;
- “Next Header” is an 8-bit field identifying the type of header (e.g. TCP, or an optional IPv6 extension header) immediately following the IPv6 header. The type values are the same IPv4’s, as defined in RFC 1700.
- Other header fields are self-explanatory.

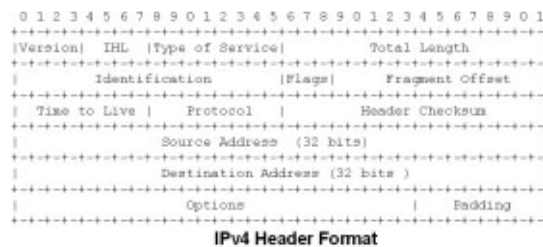


Figure 3: IPv6 vs IPv4 Header Format

Compared to an IPv4 header, shown above:

- The following fields are removed: IHL (Internet Header Length), fragmentation fields (Identification, Flags, Fragment Offset), Header Checksum;
- A Flow Label field is added;
- The following fields are replaced: Datagram Length by Payload Length, Protocol Type by Next Header, Time to Live by Hop Limit, Type of Service by Traffic Class.

IPv6 addresses are 128 bits, but IPv4 addresses are 32 bits

It is important to note that:

- IPv6 addresses are 128 bits, but IPv4 addresses are 32 bits ;
- IPv6 header length is fixed but IPv4 header length is variable with the length indicated by the IHL field. Fixed header length and no need to compute IP checksum make IPv6 forwarding simpler.

IPv6 Address Architecture

IPv6 addresses are 128-bits long. There are three types of addresses: unicast, anycast and multicast, each with its own address structure [RFC 3513]. The type of an IPv6 address is identified by the high-order bits of the address, as follows:

There are no broadcast addresses in IPv6. Their function is superseded by multicast addresses.

There are no broadcast addresses in IPv6. Their function is superseded by multicast addresses.

IPv6 Address Architecture		
Address type	Binary prefix	IPv6 notation
Unspecified	00...0 (128 bits)	::/128
Loopback	00...1 (128 bits)	::1/128
Multicast	11111111	FF00::/8
Link-local unicast	1111111010	FE80::/10
Site-local unicast	1111111011	FEC0::/10
Global unicast	(everything else)	

Unicast

IPv6 unicast addresses are aggregable with prefixes of arbitrary bit-length similar to IPv4 addresses under Classless Interdomain Routing (CIDR). Unicast addresses can be further divided into global unicast, site-local unicast, and link-local unicast. The general format for IPv6 global unicast addresses is as follows:

n bits	m bits	128-n-m bits
global routing prefix	subnet ID	interface ID

where the global routing prefix is a (typically hierarchically-structured) value assigned to a site. The subnet ID and the interface ID identify a subnet and an interface within the subnet, respectively. The address structure makes routing with IPv6 simpler. For example, a router outside a site only lookups the global routing prefix to make a routing decision.

All global unicast addresses other than those that start with binary 000 have a 64-bit interface ID field (i.e., $n + m = 64$. In fact, $n=48$ and $m=16$ in most cases). Global unicast addresses that start with binary 000 have no such constraint on the size or structure of the interface ID field. Examples of global unicast addresses that start with binary 000 are the IPv6 address with embedded IPv4 addresses. For example,

"IPv4-compatible IPv6 addresses" begin with 96 leading 0 bits followed by an IPv4 address; while "IPv4-mapped IPv6 addresses" begin with 80 leading 0 bits, then 16 1 bits (i.e. FFFF), followed by an IPv4 address.

Link-Local and Site-Local addresses are for local use only. Link-Local addresses are designed to be used for addressing on a single link for purposes such as automatic address configuration, neighbor discovery, or when no routers are present. Routers must not forward any packets with link-local source or destination addresses to other links. Site-local addresses are designed to be used for addressing inside of a site without the need for a global prefix. Routers must not forward any packets with site-local source or destination addresses outside of the site.

Anycast

An IPv6 anycast address is an address that is assigned to more than one interface (typically belonging to different systems). A packet sent to an anycast address is routed to the "nearest" interface having that address, according to the routing protocols' measure of distance. For example, multiple routers for a subnet can be assigned an anycast address. There is little experience with widespread, arbitrary use of Internet anycast addresses. Therefore, anycast addresses are currently restricted to be used on IPv6 routers only, not on hosts.

Anycast addresses are taken from the unicast address spaces and are not syntactically distinguishable from unicast addresses. A configuration command is required to turn a unicast address into an anycast address.

Multicast

Multicast addresses have the following format:

"flgs" identifies whether a multicast group is permanent or transient. "scop" identifies a multicast group's scope, e.g. site-local or global.

8 bits	4 bits	4 bits	112 bits
1111111111	flgs	scop	group ID

Riverstone's Evolving IPv6 Solution

In this section, Riverstone's IPv6 vision and the rationale behind it is explained. The phased plan itself is explained in the next section. This is done by examining what it takes for two hosts to communicate in an IPv6 context, in which at least one host is IPv6 capable. There are two possible scenarios:

1. The remote host is also IPv6 capable. The two hosts communicate with IPv6.
2. The remote host is not IPv6 capable. If both hosts are IPv4 capable, then they may communicate with IPv4. Otherwise, some protocol translation is used between IPv6 and IPv4.

This examination naturally brings out the requirements for providing an IPv6 solution. Riverstone's IPv6 plan is designed to meet such requirements in phases, based on the priority of the requirements.

IPv6 Address Acquisition

To start an IPv6 deployment, the first step is to obtain some IPv6 addresses. Carriers and service providers can obtain them from the Internet address authorities, such as APNIC in Asia, RIPE in Europe, and ARIN in the U.S. Usually, they will obtain one or multiple /32 address blocks. Enterprises obtain their IPv6 addresses from their service provider. Usually, each enterprise will get a /48 address block.

Stateless Address Auto-configuration and DHCPv6

In order to reduce the amount of configuration work, IPv6 introduces Stateless Address Auto-configuration [RFC 2462]. Routers should be able to generate link

local addresses so that hosts can auto-configure their interfaces. Alternatively, hosts can use DHCPv6 to obtain IP addresses (a.k.a “Stateful Address Auto-configuration”) or be manually configured with IPv6 addresses.

For IPv6, DNS becomes even more important. This is because, depending on the remote host’s IPv6 capability, the local host may need to communicate with either IPv6 or IPv4. DNS plays a key role in deciding which version of IP to use.

Domain Name Service (DNS)

The Domain Name System (DNS) is used in both IPv4 and IPv6 to map between names and IP addresses. For IPv6, DNS becomes even more important. This is because, depending on the remote host’s IPv6 capability, the local host may need to communicate with either IPv6 or IPv4. DNS plays a key role in deciding which version of IP to use. Each host name is associated with an IPv4 address (i.e. A record), and/or an IPv6 address (i.e. A6 record or AAAA record). This way, each host’s IPv6 capability is reflected by the existence or non-existence of an A6 or AAAA record, and is “signaled” all over the Internet via DNS. As a result, the local DNS server will know. Depending on the remote host’s IPv6 capability, the local DNS server can return an IPv6 address, or IPv4 address, or both if the remote host is dual-stacked. If the server returns both types of addresses, the local host needs to have a DNS resolver library to decide which address to use [RFC 1886].

Berkeley Internet Name Domain (BIND) version 9 is capable of handling both IPv6 and IPv4. This software distribution contains a DNS server part, named, and a client part, the resolver library. Riverstone routers or switches either serve as a transparent DNS request/reply forwarder, or a DNS client. Therefore, the resolver part of BINDv9 has a higher priority than the server part.

Neighbor Discovery

The IPv6 Neighbor Discovery protocol [RFC 2461] corresponds to a combination of the IPv4 protocols: ARP, ICMP Router Discovery and ICMP Redirect. Systems (hosts and routers) use Neighbor Discovery (ND) to determine the link-layer addresses for neighbors known to reside on attached links and to quickly purge cached values that become invalid. Hosts also use ND to find neighboring routers that are willing to forward packets on their behalf. Finally, systems use ND to actively keep track of which neighbors are reachable and which are not, and to detect changed link-layer addresses.

VRRP for IPv6

IPv6 hosts on a LAN can learn about one or more default routers by receiving Router Advertisements sent using the IPv6 Neighbor Discovery (ND) protocol. Router advertisements are multicast every few minutes. After the default router is learned, unicast ND Neighbor Solicitation messages are used to detect the failure of the default router. With ND, it will take a host 5 seconds or longer to learn that a default router is unreachable before it can switch to another default router [6], depending on ND parameter setting. This can cause some service disruption.

The Virtual Router Redundancy Protocol for IPv6 [6] (VRRP v3) specifies an election protocol that dynamically assigns the default router responsibility to one of the VRRP routers on a LAN. Using VRRP a backup router can take over for a failed default router in around 3 seconds, using VRRP default parameters.

Network Address Translation-Protocol Translation (NAT-PT)

If DNS lookup for the remote host resolves in an IPv4 address, and the local host supports only IPv6 but not IPv4, then some translation is needed between IPv6 and IPv4. NAT-PT is one such translation mechanism defined by the IETF [RFC 2767]. A dedicated NAT-PT server translates the IPv6 header for packets from the IPv6-only host to an IPv4 header, and vice versa for packets from the IPv4-only host. This translation between IPv6 and IPv4 is fundamentally similar the translation between private IPv4 and public IPv4. The limitations of NAT-PT are:

- End-to-end security, e.g. IPSec, is not possible;
- The NAT-PT server is a single point of failure.

From this perspective, NAT-PT largely defeats the purpose of using IPv6 (instead of

private IPv4 addresses). It is therefore considered undesirable.

NAT-PT translates at the network layer. There is freeware called TCP-UDP Relay that translates at the transport layer. Basically, a relay server communicates in IPv6 with the IPv6-only host, and IPv4 with the IPv4-only host, and relay between these two connections at the transport layer. TCP-UDP Relay can be used as an alternative to NAT-PT. TCP-UDP Relay has the same limitations as NAT-PT's.

Given IPv4's current dominance, it is almost certain that in the initial transition period, any host (could be an appliance such as a mobile phone) that supports IPv6 will also support IPv4. If the remote host only supports IPv4, the two hosts will simply communicate with IPv4.

Dual-Stack

For small/mid-size domains (especially universities and research institutions) that want to trial IPv6, the ideal approach is to upgrade their current IPv4 network infrastructure to support IPv6. This is more economical than adding and maintaining a separate IPv6 network. The network systems therefore need to support two control planes of IPv4 and IPv6 (a.k.a dual-stack). In addition, IPv6 must be supported over the link layer(s) used, e.g. Ethernet [RFC 2464]. Because every network link will be carrying IPv4 and IPv6 traffic simultaneously, each port on a router/switch must be able to process both types of traffic.

Tunneling

For large service providers, it can be expensive and operationally risky to upgrade all the network devices to support IPv6 simultaneously. To support IPv6 across such networks, selected edge devices are upgraded to support IPv6. IPv6 packets received at one edge device are encapsulated inside IPv4 packets and sent to appropriate remote edge devices. Obviously, such edge devices must support dual-stack. Core devices see only IPv4 packets and therefore need not support IPv6. In this environment, IPv4 addresses of the tunnel end points are specified by network operators. This is called configured tunneling. Configured tunnels are equivalent to permanent links connecting the otherwise isolated IPv6 domains. They are suitable for frequent IPv6 communication.

IPv6 over IPv4 Tunneling

This tunneling mechanism encapsulates IPv6 packets directly inside IPv4 packets. The two end points of a tunnel are manually configured. This is one of the tunneling mechanisms most widely used by service providers. MPLS can also serve as a tunneling protocol.

IPv6 over IPv4 GRE Tunneling

Compared to IPv6 over IPv4 tunneling, this tunneling mechanism adds a GRE header between the outer IPv4 header and the payload. The advantage is that in addition to IPv6 packets, i/IS-IS link state advertisements (LSAs) can also be carried. Because i/IS-IS runs over a link layer, not over IPv6, IPv6 over IPv4 GRE tunneling must be used if the two end-IPv6-domains need to run i/IS-IS over the tunnel.

MPLS

For networks that already support MPLS, MPLS may be appropriate for tunneling IPv6 traffic. In this case, the tunneling router effectively becomes a MPLS LER. The advantage of using MPLS is that it is also useful for Traffic Engineering and Virtual Private Networks (VPNs).

For networks that already support IP VPN (via RFC 2547bis), MPLS can be a good choice. RFC 2547 has already introduced a new IP address family called VPN-IPv4 that is 64 bits wide. IPv6 is effectively just another address family. Currently, there is some effort in the IETF to standardize BGP-MPLS VPN extension for IPv6 VPN [3]. With it, an IPv6 network is just like another VPN.

Tunneling IPv6 packets in IPv4 as the traverse the core may be an effective interim measure for many service providers migrating from IPv4 to IPv6



Some enterprises may want to try IPv6 before their service providers support it externally. For these IPv6 domains to communicate over an IPv4 network, they must deploy their own tunneling mechanisms, some of which are described below.

6to4 is an automatic tunneling mechanism. For any enterprise that has a public IPv4 address, 6to4 tunneling [31] can be used. By prefixing a 0x2002::/16 to the public IPv4 address of the enterprise's access link to its SP, the enterprise will automatically obtain a globally unique /48 IPv6 prefix. The enterprise can then do normal IPv6 routing inside its domain.

6to4 Tunneling

6to4 is an automatic tunneling mechanism. For any enterprise that has a public IPv4 address, 6to4 tunneling [31] can be used. By prefixing a 0x2002::/16 to the public IPv4 address of the enterprise's access link to its SP, the enterprise will automatically obtain a globally unique /48 IPv6 prefix. The enterprise can then do normal IPv6 routing inside its domain. Such an enterprise is called a 6to4 site. The only place in a 6to4 site that 6to4 tunneling needs to be deployed is its border router to its SP. When the border router receives an IPv6 packet to another 6to4 site, it will encapsulate the IPv6 packet inside an IPv4 packet, and use the public IPv4 address inside the destination IPv6 address as the IPv4 destination address. This IPv4 packet will be sent via the IPv4 Internet to the destination 6to4 site's border router, where the IPv4 packet will be decapsulated and the resultant IPv6 packet will be routed via IPv6 to the destination host.

Hosts in a 6to4 site need not implement 6to4 tunneling. But as described in the DNS section, they need to be able to select proper destination IPv6 address for the receiver, as that will determine whether 6to4 tunneling will be used. If one host (either the sender or the receiver) has only a 6to4 address, and the other one has both a 6to4 and a native IPv6 address, then the 6to4 address should be used for both. If both hosts have a 6to4 address and a native IPv6 address, then either the 6to4 address should be used for both, or the native IPv6 address should be used for both. The choice should be configurable. The default configuration should be native IPv6 for both. And in this case, native IPv6 routing (rather than 6to4 tunneling) will be used.

6to4 tunneling is simple and straightforward. It is the most widely used tunneling protocol for IPv6 in enterprises.

IPv4-compatible Tunneling

IPv4-compatible tunneling is another automatic tunneling mechanism. If the end point is the destination host, and the destination host has an IPv4-compatible IPv6 address, then IPv4-compatible tunneling can be done, as described in [RFC 2893]. An IPv4-compatible IPv6 address of a host is formed by prepending a 0/96 prefix to the host's public IPv4 address. With IPv4-compatible tunneling, a device infers the IPv4 address of the tunnel end point from the IPv4-compatible IPv6 address, encapsulates the IPv6 packet inside an IPv4 packet (with the inferred IPv4 address as the destination IP address), and sends it to the tunnel end point. No manual tunnel configuration is needed. Network operation and management is thus simplified.

IPv4-compatible tunneling is useful for systems (hosts or routers) with IPv4-compatible IPv6 addresses to communicate with each other. This is suitable for isolated IPv6 hosts with IPv4-compatible IPv6 addresses. But if such systems want to communicate with systems without IPv4-compatible IPv6 addresses, they need a configured tunnel to a router so that the router can route packets to their final destinations. The following picture illustrates such a deployment scenario: an host with IPv4-compatible IPv6 address (the left one) use IPv4-compatible tunneling to communicate with another host with IPv4-compatible IPv6 address (the top one), and use a configured tunnel to a router to communicate with a host with native IPv6 address (the right one). The inconvenience to communicate with non-IPv4-compatible systems is one of the limitations of IPv4-compatible tunneling. Furthermore, tunneling is between 2 end systems (as opposed to 2 domains with 6to4 tunneling), and each IPv4-compatible end system needs a public IPv4 address (as opposed to 1 public IPv4 address per domain). IPv4-compatible tunneling therefore does not scale well.

Tunnel Broker

A Tunnel broker can be used to reduce manual tunnel configuration work, especially for the administrators of enterprise networks. A service provider provides a web server equipped with tunnel configuration scripts. When an end host in an enterprise needs a tunnel to an IPv6 network, the end host accesses the web server to enter some information and run the script designed for that enterprise. The script will configure the end host and the other end of the tunnel, usually a SP router, to set up the tunnel. The web server is called a tunnel broker. All communications between a tunnel broker and the two tunnel end points are IPv4 based.

Since all a tunnel broker does is providing the scripts, there is no additional implementation needed from an equipment vendor. A carrier can write scripts to

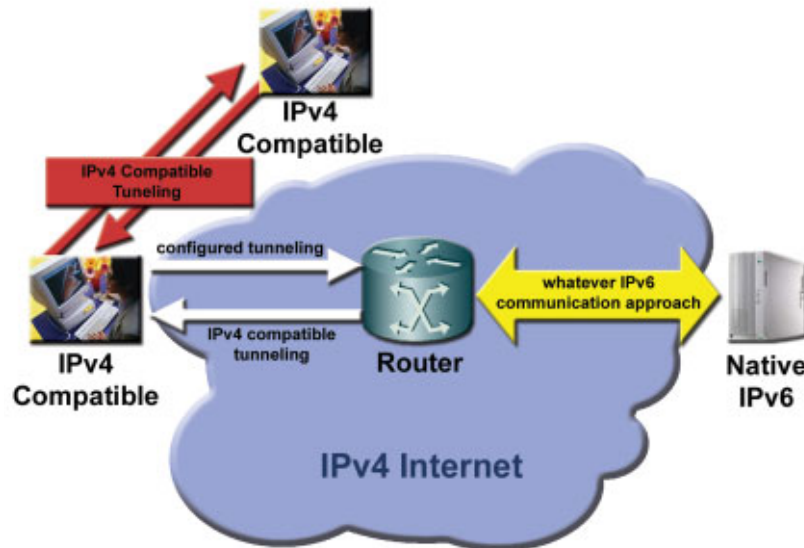


Figure 4: IPv6 Interoperating with IPv4

activate the tunneling mechanism provided by Riverstone Networks so that they can provide a tunnel broker to their customers. Because the end hosts and tunneling routers are open to outside scripts, security issues should be considered.

Miscellaneous requirements related to tunneling

In addition to adding an IPv4 header, the encapsulating device also needs to handle some tunneling-related issues:

- Determine when to fragment and when to report an ICMP "packet too big" error back to the source. Note that what is discussed above is IPv4 fragmentation. It is worth noting that IPv6 itself mandates that all links in the Internet support a minimal MTU of 1280 Bytes. Therefore, a minimal IPv6 implementation can simply send packets smaller than 1280 Bytes and avoid implementing IPv6 MTU discovery. In addition, IPv6 fragmentation, if any, only happens at the source, not at any intermediate system. (But IPv4 fragmentation can happen anywhere). IPv6 MTU discovery only serves to improve efficiency by being able to send larger packets without causing fragmentation. Nonetheless, field input indicates that research institutes are interested in this feature (so that they can use jumbo frames whenever possible).
- Translate IPv4 ICMP "packet too big" errors received from downstream routers along the tunnel back to the source into IPv6 ICMP "packet too big" errors [RFC 2893]. This is a medium priority task.

In addition, both configured and automatic tunnels are IPv6 interfaces (over the IPv4 “link layer”). They must have link-local addresses. The link-local addresses are used by routing protocols operating over the tunnels.

Similar to an IPv4 domain, an IPv6 domain needs some routing protocols so that routers/switches can communicate with each other. These include Interior Gateway Protocols (IGPs) for internal communication and Exterior Gateway Protocols (EGPs) for communication with an external IPv6 domain.

While both i/IS-IS and OSPF v3 are industrial strength IGPs suitable for networks of most any size, i/IS-IS will likely be easier to implement.

IGP

The IGP of choice can be RIPng, i/IS-IS or OSPF v3. They are discussed below.

- RIPng [RFC 2080]

RIPng is effectively RIP v3 with IPv6 support. RIP is a distance vector protocol. It is simple, easy to implement, but has some limitations. For small and simple enterprise networks, RIPng may be sufficient.

- i/IS-IS [4]

i/IS-IS is an industry-strength, link state protocol that is suitable for almost any networks. Because of the way the protocol is defined and the use of Type-Length-Value (TLV), i/IS-IS is relatively easy to implement.

- OSPF v3 [RFC 2740].

OSPF v3 is also an industry-strength, link state protocol that is suitable for almost any network. Some protocol modifications need to be made to OSPF v2. OSPF v3 is relatively difficult to implement.

Many enterprise networks run OSPF v2 so OSPF v3 would seem to be the natural choice for IPv6. However, OSPF v3 is quite different from OSPF v2, thus most enterprises will find little difference in the OSPF v3 and i/IS-IS learning curves.

EGP

Multi-protocol BGP [RFC 2858] is the protocol of choice.

Policy based routing

Policy based routing essentially means making routing decisions based on information other than destination IP addresses. IPv6 addresses are well structured and can convey more information than IPv4 addresses, which reduces the need for additional information not carried in destination addresses. IPv6 addresses can also be aggregated more easily. Therefore, the use of IPv6 can reduce dependence on policy based routing. Nonetheless, service providers generally need some mechanism to explicitly control routing.

Multicast

Multicast Listener Discovery (MLD) [RFC2710], the equivalent of IGMPv2 for IPv6, and MLD v2 (MLDv2) [7], the equivalent of IGMPv3, are considered important by many service providers. For multicast routing, PIM-SM and MSDP v2 are useful for IPv6 LAN, and PIM-SSM is useful for IPv6 WAN.

ACLs and Security

Just like IPv6 simplifies policy based routing, it also simplifies the use of ACLs. It is a given that those deploying IPv6 will want to have control on traffic and security. Without the NAT barrier, IPSec will be end-to-end, which is very desirable. ACL and security mechanisms for IPv6 and IPv4 are fundamentally similar. However, because IPv6 addresses are 128-bit, ACL lookup keys become very long, which leads to a very wide ACL lookup table and poses some implementation challenges. In addition,

because TCP/UDP header's location in a IPv6 packet is not fixed, any L4 (or higher-layered) filters will be difficult to implement.

In a dual-stack environment, a system is susceptible to both IPv4 and IPv6 types of DoS attacks. This makes security mechanisms even more important. IPSec is used more often in an IPv6 environment than in an IPv4 one.

Many research institutes have deployed IPv6 without ACLs because of lack of support from vendors., a situation expected to change rapidly in the near future.

QoS for both IPv4 and IPv6 is usually based on Diffserv.

QoS

The current QoS mechanism for IPv4 is based on Diffserv and its extension for MPLS. Diffserv is defined for both IPv6 and IPv4. This means that IPv6 can use the same QoS mechanism as IPv4 – only with different header fields.

Network management

Network management is an essential part of any IP solution. Two major tasks of network management are:

- Network monitoring and troubleshooting, e.g. ICMP ping and traceroute for IPv6
- Statistics collecting and reporting: SNMP and MIBs

IPv6 network management capabilities are considered extremely useful and badly needed, with IP Forwarding Table MIB and MIB for IP being the two most important MIBs.

L2 Support for IPv6

Using a protocol-based VLAN mechanism, it is possible to separate IPv6 traffic from other network layer traffic (e.g. IPv4 or IPX), sending it on to separate VLANs or VPLS instances. This way, IPv6 systems can be connected via IPv6 VLANs or VPLS instances and have their own broadcast domain. The advantage of this approach is that it provides a very simple way for an enterprise, to connect IPv6 systems. But because protocol-based VLAN mechanism separates different protocol traffic based on EtherType, and there can be many EtherTypes belonging to the same protocol (e.g. EtherType 0800x and 0806x all carry IPv4 traffic), the enumeration of EtherType may be complex. Asking an enterprise to present different types of traffic in different VLANs to their service providers would be the cleanest solution. And in this case, the service providers will be providing a simple L2 service without relying on protocol-based VLAN.

Riverstone has a 3 Phase Approach to IPv6 implementation, starting with more basic, fundamental features and incrementally adding more advanced feature support in a manner that mirrors most common IPv6 implementation roadmaps.

IPv6 – Riverstone's Phased Approach

Riverstone's next generation products are all designed from ground up to support IPv6. They share a common network operating system, ROS-X, which will support a complete IPv6 solution.

Phase 1

Currently, the 15008 fully supports IPv6 data plane functionalities. On the control plane, ROS-X supports static routing.

Phase 2

In Release 2.0, ROS-X provides the fundamental IPv6 dynamic routing and network management functionalities. The goal of phase-2 is to enable customers to conduct proof of concept or labs trials of basic commercial IPv6 services. The major features supported in 2.0 include:

- DNS for IPv6
- DHCP v6
- Dual protocol stacks (IPv4 and IPv6)
- Neighbor Discovery
- Dynamic routing: i/IS-IS, MP-BGP, policy based routing
- ACLs for IPv6



- MPLS tunneling of IPv6
- Management: SNMP/MIBs, ICMP v6, IPv6 path MTU discovery

Phase 3

After Release 2.0, ROS-X will provide a comprehensive IPv6 solution. The goal of phase-3 is to enable customers to deploy sophisticated commercial IPv6 services. The major additional features we plan to support include:

- VRRP v3
- IPv6 over IPv4 tunneling
- IPv6 over IPv4 GRE tunneling
- 6to4 tunneling
- OSPF v3
- IPv6 QoS
- IPv6 multicast
- Stateless Address Auto-configuration
- IPv6 over POS links
- BGP-MPLS VPN extension for IPv6 VPNs
- IPv4-compatible tunneling

Conclusion

Riverstone's next generation products and modular network operating system, ROS-X, were designed and optimized for IPv6 from the ground up. As mobile applications, online gaming, IP TV, home networking and demand from Asia remove the chicken and egg IPv6 Application/Infrastructure logjam, Riverstone's 15000 family is uniquely positioned to help service providers deliver profitable, carrier class IPv6 based services. Redundant hardware, NEBS compliant design, modular software and custom ASICs help Riverstone deliver on the promise of robust, scalable, rich commercial services, allowing carriers to hit triple play home runs and their customers to enjoy the quality of experience that they have come to expect.

References

1. CNet News, http://news.com.com/China+to+trump+U.S.+in+broadband+subscribers/2100-1034_3-5695591.html
2. Internet draft, IPv6 Node Requirements
3. Internet draft, BGP-MPLS VPN extension for IPv6 VPN
4. Internet draft, Routing IPv6 with IS-IS
5. Internet draft, Dynamic Host Configuration Protocol for IPv6 (DHCPv6)
6. Internet draft, Virtual Router Redundancy Protocol for IPv6
7. Internet draft, Multicast Listener Discovery Version 2 (MLDv2) for IPv6
8. IPv6 Forum, <http://www.ipv6tf.org/pdf/ISTClusterbooklet2005.pdf>.
9. RFC 1886, DNS Extensions to Support IP version 6
10. RFC 1981, Path MTU Discovery for IP version 6
11. RFC 2080, RIPng for IPv6
12. RFC 2374, An Aggregatable Global Unicast Address Format
13. RFC 2460, Internet Protocol, Version 6 (IPv6) Specification
14. RFC 2461, Neighbor Discovery for IP Version 6 (IPv6)
15. RFC 2462, IPv6 Stateless Address Autoconfiguration
16. RFC 2463, Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification
17. RFC 2464, Transmission of IPv6 Packets over Ethernet Networks
18. RFC 2472, IP Version 6 over PPP
19. RFC 2473, Generic Packet Tunneling in IPv6 Specification
20. RFC 2474, Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers
21. RFC 2492, IPv6 over ATM Networks
22. RFC 2545, Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing
23. RFC 2590, Transmission of IPv6 Packets over Frame Relay Networks Specification
24. RFC 2640, Internet Protocol, Version 6 Specification
25. RFC 2710, Multicast Listener Discovery (MLD) for IPv6 (not supported by Cisco)
26. RFC 2740, OSPF for IPv6
27. RFC 2765, Stateless IP/ICMP Translation Algorithm (SIIT)
28. RFC 2766, Network Address Translation-Protocol Translation (NAT-PT)
29. RFC 2858, Multiprotocol Extensions for BGP-4

30. RFC 2893, Transition Mechanisms for IPv6 Hosts and Routers
31. RFC 3056, Connection of IPv6 Domains via IPv4 Clouds
32. RFC 3068, An Anycast Prefix for 6to4 Relay Routers
33. RFC 3513, IP Version 6 Addressing Architecture Specification
24. RFC 2640, Internet Protocol, Version 6 Specification
25. RFC 2710, Multicast Listener Discovery (MLD) for IPv6 (not supported by Cisco)
26. RFC 2740, OSPF for IPv6
27. RFC 2765, Stateless IP/ICMP Translation Algorithm (SIIT)
28. RFC 2766, Network Address Translation-Protocol Translation (NAT-PT)
29. RFC 2858, Multiprotocol Extensions for BGP-4
30. RFC 2893, Transition Mechanisms for IPv6 Hosts and Routers
31. RFC 3056, Connection of IPv6 Domains via IPv4 Clouds
32. RFC 3068, An Anycast Prefix for 6to4 Relay Routers
33. RFC 3513, IP Version 6 Addressing Architecture Specification
24. RFC 2640, Internet Protocol, Version 6 Specification
25. RFC 2710, Multicast Listener Discovery (MLD) for IPv6 (not supported by Cisco)
26. RFC 2740, OSPF for IPv6
27. RFC 2765, Stateless IP/ICMP Translation Algorithm (SIIT)
28. RFC 2766, Network Address Translation-Protocol Translation (NAT-PT)
29. RFC 2858, Multiprotocol Extensions for BGP-4
30. RFC 2893, Transition Mechanisms for IPv6 Hosts and Routers
31. RFC 3056, Connection of IPv6 Domains via IPv4 Clouds
32. RFC 3068, An Anycast Prefix for 6to4 Relay Routers
33. RFC 3513, IP Version 6 Addressing Architecture

