



IPv6: Providing IPv6 Services over an IPv4 Backbone Using Tunnels

Version History

Version Number	Date	Notes
1	01 July 2002	This document was created.
2	19 May 2003	Updated the related documents section.

This document explains how an Internet service provider (ISP) can provide basic, new IP version 6 (IPv6) services to customers by using manually configured IPv6 overlay tunnels across its existing IPv4 network infrastructure.

As IPv6 grows in popularity and the benefits of IPv6 are realized, ISPs are interested in integrating IPv6 into their existing IPv4 networks so that they can validate IPv6 products and assess future demand for IPv6 services before making financial and resource commitments toward deploying IPv6 throughout their network infrastructures.

This document is intended for ISPs that are familiar with IPv6 and IPv6 overlay tunneling techniques, and have experience with Cisco Systems networking equipment and the Cisco IOS software. This document explains the use of IPv6 tunnels in the context of an ISP providing IPv6 services to its customers. This document does not provide an in-depth explanation of IPv6 tunnel types or the costs associated with implementing each type of tunnel. Additionally, this document does not discuss providing IPv6 services over dedicated Layer 2 connections, such as ATM or Frame Relay permanent virtual circuits (PVCs). For more information on IPv6 and IPv6 overlay tunneling techniques, refer to the IPv6 documents and the IPv6 for Cisco IOS Software feature documentation listed in the “[Related Documents](#)” section on page 18.

This document includes the following sections:

- [Business Objectives, page 2](#)
- [Possible Solutions, page 3](#)
- [Proposed Solution: Manually Configured IPv6 Overlay Tunnels, page 4](#)
- [Implementation, page 9](#)
- [Related Documents, page 18](#)

Business Objectives

An intermediate production ISP believes that adding basic IPv6 services to its suite of service offerings will attract customers and increase its overall business. Examples of basic IPv6 services are connectivity to the IPv6 Internet (which includes access to both the IPv6 production Internet and the 6BONE) and hosting IPv6 web, IPv6 Internet Relay Chat (IRC), and IPv6 Internet gaming servers.

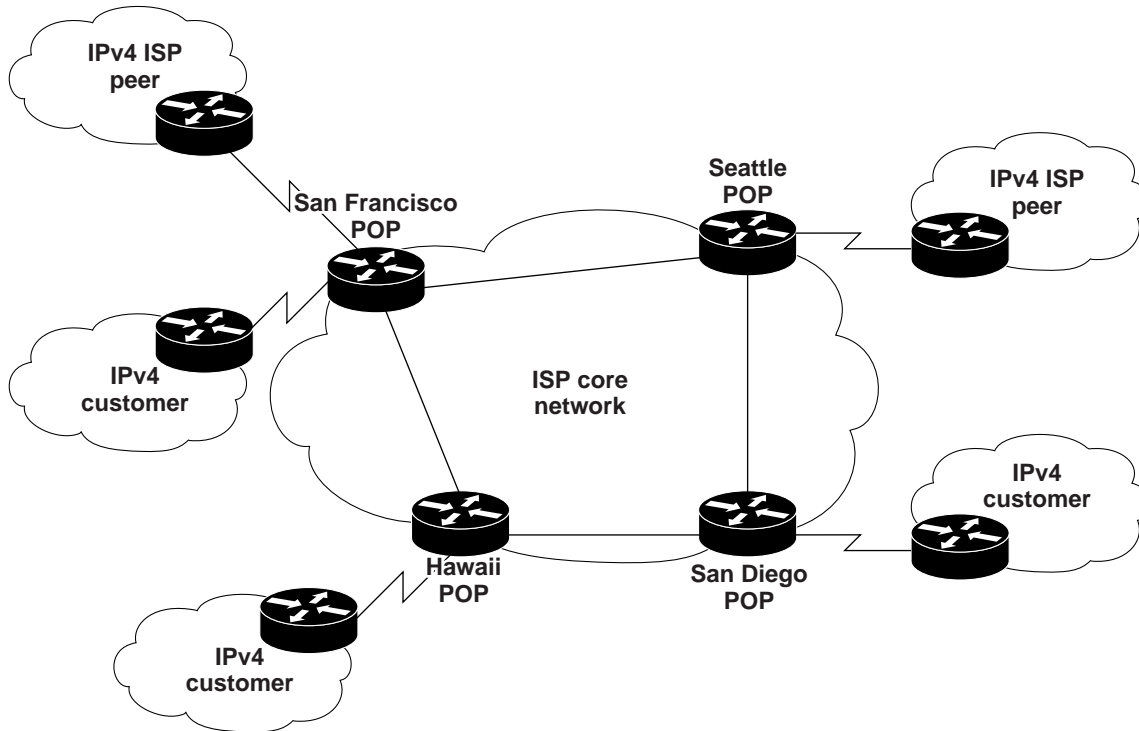
Following are the ISP business objectives addressed in this document:

- With the smallest financial and resource commitment possible, integrate IPv6 products into the existing IPv4 topology to validate the products and assess future demand for IPv6 services.
- Create new revenue streams based on newly offered IPv6 services.
- Demonstrate IP leadership in the industry by offering IPv6 services to early adopters.

Initial Network Topology

Figure 1 shows the existing IPv4 topology of the ISP, which includes the IPv4 connections to two ISP peers and three customer sites. Four partially meshed border routers interconnect four ISP points of presence (POPs) in Seattle, San Francisco, San Diego, and Hawaii. The border routers in the Seattle and San Francisco POPs have dedicated WAN connections to two different ISP peers. The border routers in the San Francisco, San Diego, and Hawaii POPs have a dedicated WAN connection to the border routers of three separate customer sites.

Figure 1 Initial ISP IPv4 Network Topology



80292

Possible Solutions

The ISP can deploy manually configured IPv6 tunnels or a 6to4 relay service (which uses 6to4 tunnels) to integrate IPv6 products into its existing IPv4 topology and provide IPv6 services to customers. Manually configured IPv6 tunnels and 6to4 tunnels require that the host or router at each end of the tunnel support both the IPv4 and IPv6 protocol stacks (a host or router that supports both protocol stacks is considered to be dual-stacked). The Cisco IOS software supports manually configured IPv6 tunnels, 6to4 tunnels, and dual-stacked hosts.

**Note**

Another possible solution would be for the ISP to upgrade all of its routers (at the core, distribution, and customer access levels of its network) to be dual-stacked and to provide IPv6 services to its customers over dedicated, IPv6-native connections. This solution is not addressed in this document because the ISP wants to evaluate IPv6 products and the demand for IPv6 services before it moves forward with the substantial financial and resource commitment of upgrading its entire network to be dual-stacked. Providing IPv6 services to customers by deploying IPv6 overlay tunnels requires that the ISP upgrade only a few of its routers to be dual-stacked.

Possible Solution #1: Manually Configured IPv6 Overlay Tunnels

IPv6 overlay tunnels encapsulate IPv6 packets within IPv4 packets (IPv6 is the passenger protocol and IPv4 is the encapsulation protocol). A manually configured IPv6 tunnel is a technique where an IPv6 address is manually configured on the tunnel interface and IPv4 addresses are manually configured at the tunnel source and the tunnel destination.

Manually configured IPv6 tunnels must be requested and negotiated between customers (one tunnel for each customer) and the ISP. As a result, manually configured IPv6 tunnels enable the ISP to connect and disconnect each customer independently, enforce policies (control the traffic across each tunnel), and charge for services on a per-customer basis. Additionally, manually configured IPv6 tunnels provide stable, secure communication between the tunnel endpoints (with unique traffic statistics for each tunnel) and enable the ISP to delegate a /48 prefix to each customer (from the /40 prefix of the applicable POP). Each customer must use the delegated /48 prefix to create the IPv6 addressing scheme for its site. By delegating a /48 prefix to each customer site, the ISP is free to change its service offerings (for example, offering only native IPv6 services) without affecting the numbering scheme or tunnels of each customer site (the sites and tunnels need not be renumbered at each ISP service change).

Although manually configured IPv6 tunnels provide the ISP with a great amount of control over its customer connections, the one-to-one relationship between each customer and a tunnel means the maintenance and management of the tunnels increases dramatically as the ISP adds customers. As the number of customers increases, the IPv6 address assignments within each POP should be managed with the understanding that additional routers may be required to accommodate the number of manually configured IPv6 tunnels (each router type can support only a finite number of tunnels, the number of which is dictated by the type of CPU and the amount of DRAM installed in the router).

Possible Solution #2: 6to4 Relay Service (Using 6to4 Overlay Tunnels)

A 6to4 tunnel is an automatic IPv6 tunnel where a 6to4 border router in an isolated IPv6 network creates a tunnel to a 6to4 border router in another isolated IPv6 network over an IPv4 infrastructure. The tunnel destination is determined by the globally unique, 32-bit IPv4 address of the remote 6to4 border router that is concatenated to the prefix 2002::/16. 6to4 tunnels are configured between 6to4 border routers or between 6to4 border routers and hosts.

A 6to4 relay service is a 6to4 border router that offers traffic forwarding to the IPv6 Internet for remote 6to4 border routers. A 6to4 relay forwards packets that have a 2002::/16 source prefix.

6to4 tunnels and connections to a 6to4 relay service need not be requested or negotiated between customers and the ISP. The ISP simply configures the 6to4 relay service and customers can automatically connect to the service whenever they like. Because of the one-to-many relationship between the 6to4 relay service and each 6to4 tunnel (each customer), there is low maintenance and management overhead associated with 6to4 tunnels and a 6to4 relay service. However, given that customers use the IPv4 address of their border router to construct the 6to4 address that they use to connect to the 6to4 relay service (they are not delegated a /48 prefix from the ISP), the ISP may want to manage the IPv4 routing announcements for the relay service to control its use (the ISP will need IPv4 traffic statistics if it wants to identify and charge individual customers for using the service).

Proposed Solution: Manually Configured IPv6 Overlay Tunnels

The proposed solution is to provide IPv6 services to customers by deploying manually configured tunnels because the tunnels combine stable, secure communication with the flexibility of enforcing policies and billing for services on a per-tunnel basis.

Strategy

Provide the IPv6 service at the customer access level by tunneling IPv6 traffic over the existing IPv4 infrastructure of the ISP. Starting at the customer access level enables the ISP to offer an IPv6 service immediately without a major upgrade to its network infrastructure and without impacting its existing IPv4 services. This strategy also enables the ISP to evaluate IPv6 products and assess demand for IPv6 services before deploying IPv6 throughout its network infrastructure.

Based on the geographical location of the customers requesting IPv6 services and the availability of support staff, identify the border routers within the existing IPv4 infrastructure to be dual-stacked. ISP POPs that have many local customers requesting IPv6 services should have their border routers upgraded to be dual-stacked; however, these POPs will need to have available staff to support the IPv6 customers.



Note

In most cases, only a Cisco IOS software upgrade is needed on the identified border routers to support a dual-stack configuration and the manually configured IPv6 tunnels; new hardware need not be provisioned. Refer to the IPv6 for Cisco IOS Software feature documentation in the New Features in Release 12.2 T or New Features in Release 12.0 ST areas of Cisco.com for IPv6 supported platform information. If you have an account on Cisco.com, you can use Cisco Feature Navigator to display IPv6 supported platform information. The Cisco Feature Navigator home page is located at the URL <http://www.cisco.com/go/fn>.

Prevent IPv6 from leaking into the existing IPv4 infrastructure by enabling IPv6 routing only on router interfaces used to route traffic over the IPv6 tunnels and native IPv6 router interfaces.



Note

Currently, the Cisco IOS software does not support the re-marking of tunneled IPv6 packets with IPv6 quality of service (QoS) labels. IPv6 packets marked with an IPv6 QoS label are automatically mapped to the class of service defined for IPv4 traffic of the same type when the IPv6 packets are encapsulated in IPv4 packets by a dual-stack router. For example, IPv6 packets with a type of service (ToS) precedence of 5 are automatically mapped to the priority queue for IPv4 packets with a precedence of 5 when the IPv6 packets are routed through a tunnel by a dual-stack router.

Apply for an IPv6 prefix from the applicable regional registry.

**Note**

The solution presented in this document is based on a North American ISP that is assigned a /32 IPv6 prefix from the American Registry for Internet Numbers (ARIN), which assigns IPv6 prefixes in North America.

During preparation of this document, ARIN was assigning /32 IPv6 prefixes. We urge you to contact ARIN directly for information on its current IPv6 prefix assignment policy, which is subject to change.

From the /32 prefix, the ISP assigns a /40 IPv6 prefix to each of its POPs and a /48 IPv6 prefix to each of its customer sites. However, the IPv6 and IPv4 addresses used in the example configurations in this document are not globally routable and are provided for illustrative purposes only.

Using the IPv6 prefix from the registry, perform the following tasks:

- Delegate a /40 IPv6 prefix to the Seattle and San Francisco POPs.
- Delegate a /48 IPv6 prefix from the /40 IPv6 prefix of the San Francisco POP to the customer site that is requesting IPv6 services.
- Configure the border routers in the San Francisco and Seattle POPs to be dual-stacked.
- Configure a dedicated, IPv6-native connection between the border routers in the San Francisco and Seattle POPs.
- Configure a manual IPv6 tunnel with IPv4 as both the encapsulation and transport protocol between the border router in the Seattle POP and the border router in the IPv6 exchange point in Chicago.

**Note**

Multiprotocol BGP is the exterior gateway protocol used between the Seattle POP and the IPv6 exchange point.

Given that all IPv6 Internet traffic from the ISP is routed through the tunnel to the IPv6 exchange point, the solution presented in this document does not require route maps on the ISP border routers to filter IPv6 prefixes. The ISP would need to deploy route maps if it decided to establish a second connection to the IPv6 internet through a peering relationship with another IPv6 exchange point (the route maps would keep the ISP network from being used as a transit network between the two exchange points).

- Configure a manual IPv6 tunnel with a static route between the border router in the San Francisco POP and the border router of the customer site in Hawaii that is requesting IPv6 services.

**Note**

Given that the border router in the Hawaii POP was not upgraded to be dual-stacked, IPv6 traffic from the customer site in Hawaii must be tunneled through the Hawaii POP to the border router in the San Francisco POP.

Network Topology

Figure 2 shows the addition of the customer requesting IPv6 services and the IPv6 exchange point to the existing ISP network topology. Specifically, an existing IPv4 customer that is connected to the Hawaii POP is now requesting access to the IPv6 Internet, which is accessible through the IPv6 exchange point in Chicago. The border routers in the Seattle and San Francisco POPs have been configured to be dual-stacked.


Note

The border routers in the San Diego and Hawaii POPs were not upgraded to be dual-stacked because the great majority of the ISP support staff is located in the San Francisco and Seattle POPs, and the ISP anticipates that the majority of IPv6 service requests will originate from customers located in San Francisco and Seattle.

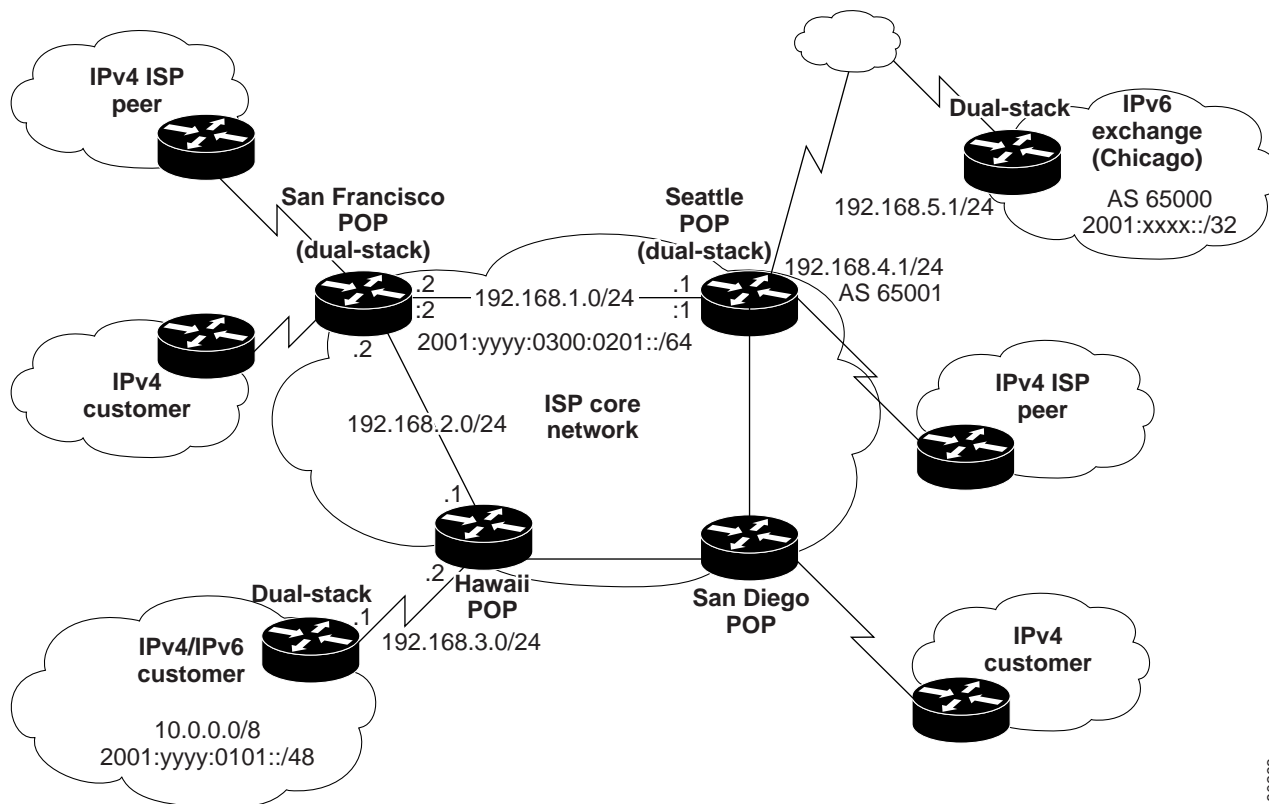
The IPv6 exchange point has been delegated the IPv6 prefix 2001:xxxx::/32 from the regional registry (ARIN). The ISP has been delegated the IPv6 prefix 2001:yyyy::/32 from the regional registry. The border routers in the Seattle POP, San Francisco POP, IPv6 exchange point, and customer site have been configured to be dual-stacked. Additionally, the IPv6 exchange point is in autonomous system (AS) 65000 and the Seattle POP is in AS 65001. The Seattle and San Francisco POPs have been delegated the prefixes 2001:yyyy:0300::/40 and 2001:yyyy:0100::/40, respectively, from the IPv6 prefix 2001:yyyy::/32. The customer site requesting IPv6 services has been delegated the prefix 2001:yyyy:0101::/48, which stems from the 2001:yyyy:0100::/40 prefix of the San Francisco POP.


Note

The 2001:xxxx::/32 and 2001:yyyy::/32 prefixes used in the example configurations in this document are not globally routable and are provided for illustrative purposes only.

AS 65000 and AS 65001 are from the range of private autonomous system (AS) numbers (AS 64512 through to AS 65535) designated by the Internet Assigned Number Authority (IANA) and are used in this document for illustrative purposes only.

Figure 2 Postimplementation Network Topology—IP Addressing



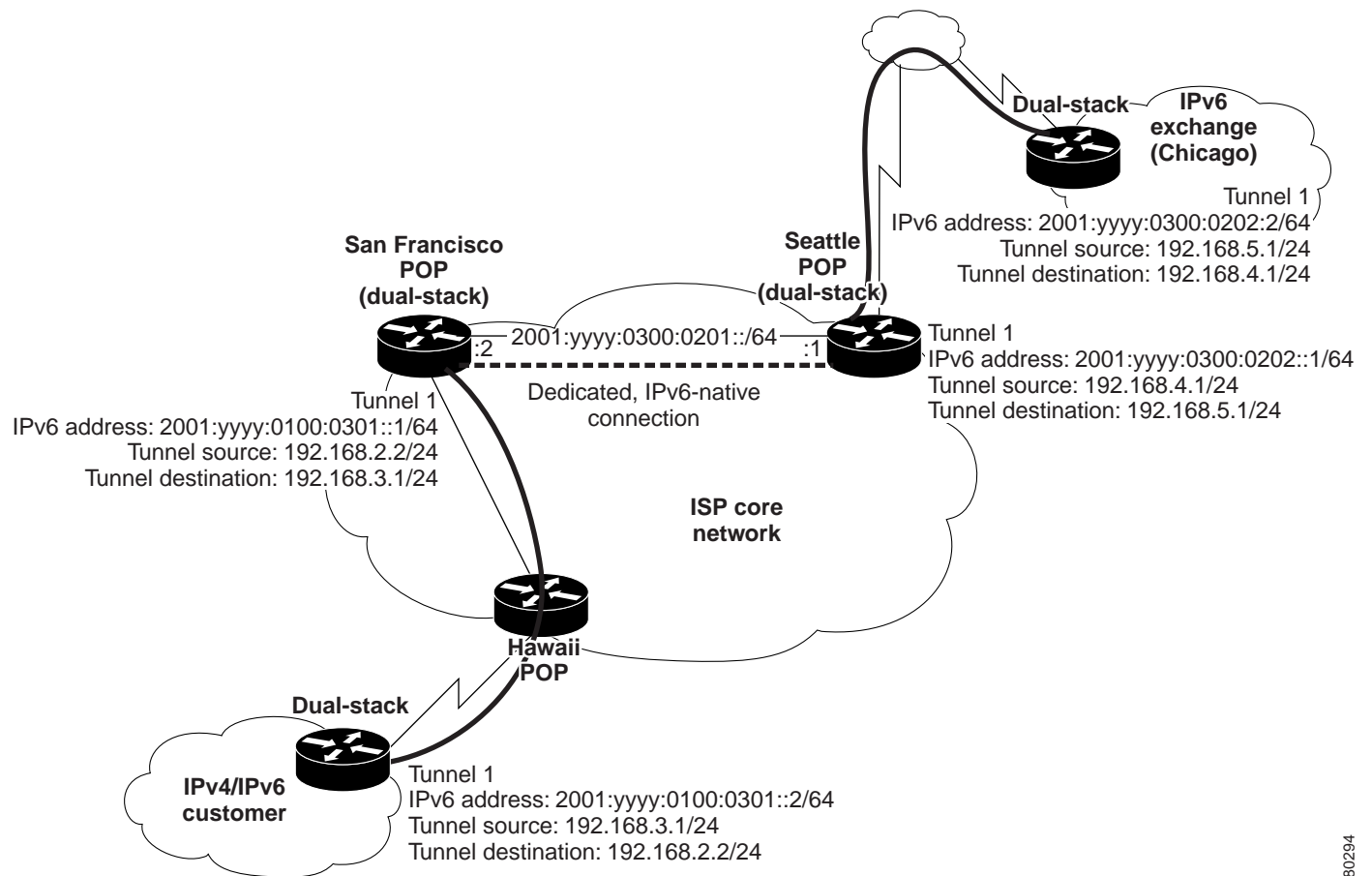
80293

A dedicated, IPv6-native connection is configured between the Seattle and San Francisco POPs. (See [Figure 3](#).) A manually configured IPv6 tunnel with a static route is configured between the border router in the San Francisco POP and the border router in the customer site. Another manually configured IPv6 tunnel is configured between the border router in the Seattle POP and the border router in the IPv6 exchange point. Both tunnels are configured over the existing IPv4 connection between the ISP and customer, and between the ISP and the IPv6 exchange point (using globally routable IPv4 addresses).



Note For clarity, the San Diego POP and the IPv4-only customers of the ISP have been removed from [Figure 3](#).

Figure 3 Postimplementation Network Topology—IPv6 Manually Configured Tunnels



80294

How This Solution Works

A node in the customer site generates IPv6 packets (for example, by initiating an IPv6 Internet gaming session) destined for the IPv6 Internet (which is accessible through the IPv6 exchange point in Chicago). After reaching the border router of the IPv6 customer site, the IPv6 packets are encapsulated within IPv4 packets and routed (by using a static route) over the manually configured IPv6 tunnel—through the Hawaii POP border router—to the San Francisco POP border router. The Hawaii POP border router processes the encapsulated IPv6 packets as ordinary IPv4 packets. At the San Francisco POP border router, the IPv4 encapsulation is stripped from the IPv6 packets and the IPv6 packets are sent across the dedicated IPv6 link to the Seattle POP border router. After arriving at the Seattle POP border router, the IPv6 packets are encapsulated within IPv4 packets and routed (by using multiprotocol BGP extensions for IPv6 as the exterior gateway protocol) over another manually configured IPv6 tunnel to the IPv6 exchange point border router and, ultimately, to the IPv6 Internet.

Benefits

Following are the benefits of the ISP using manually configured IPv6 tunnels to provide IPv6 services to customers:

- Provides IPv6 connectivity to IPv6 customer sites without disrupting the existing IPv4 service of the ISP and without a large financial or resource commitment from the ISP.
- Creates new revenue streams for the ISP by offering new IPv6 services and support for new IPv6 applications.
- Enables the ISP to validate and gauge future demand for IPv6 while generating revenue from initial customer sites that request IPv6 services.
- Provides stable, secure connections between the ISP, the customer site, and the IPv6 exchange point.
- Enables the ISP to monitor individual traffic statistics for the customer site, and charge the customer site for IPv6 services.
- Enables the ISP to change its IPv6 service offerings without affecting the /48 IPv6 prefix delegated to the customer site.

Ramifications

Following are the ramifications of the ISP using manually configured IPv6 tunnels to provide IPv6 services to customers:

- Supporting additional IPv6 customers will require the ISP to manually configure additional IPv6 tunnels. Management overhead increases as the number of tunnels increases.
- At some point, the number of manually configured IPv6 tunnels that each ISP border router can support might be exhausted and the ISP will need to upgrade the border router hardware. (The ISP will need to reengineer its IPv6 service offering).



Note

To improve its IPv6 service offering, the ISP could move the IPv6 tunneled traffic to dedicated Layer 2 connections, such as ATM or Frame Relay PVCs, or upgrade its entire network to be dual-stacked. However, in most cases, an ISP would need to compare the revenue generated from IPv6 services against the operational costs of providing the services before moving or upgrading its network.

Implementation

This section describes the required tasks for an ISP to provide IPv6 services to customers by deploying manually configured tunnels. It contains the following sections:

- [Prerequisites and Design Considerations](#)
- [Implementation Process Steps](#)
- [Device Characteristics and Annotated Configuration Files](#)

Prerequisites and Design Considerations

Before providing IPv6 services to customers by deploying manually configured tunnels, the ISP must perform the following tasks:

- Identify the border routers within the existing IPv4 infrastructure to be dual-stacked. When identifying the border routers, consider the following items:
 - Each border router within the ISP network (for example, the border routers that interconnect the Seattle POP, San Francisco POP, and the customer site in Hawaii) must have a static IPv4 address that is routable within the ISP network and available for use in IPv6 tunnel configurations. Additionally, each border router that connects the ISP network to another network (for example, the border routers that interconnect the Seattle POP and the IPv6 exchange point in Chicago) must have a static IPv4 address that is globally routable and available for use in IPv6 tunnel configurations.



Note

The IPv4 addresses used in the example configurations in this document are not globally routable and are provided for illustrative purposes only.

- Each border router can support a finite number of manually configured IPv6 tunnels. (The number of supported tunnels is dictated by the type of CPU and the amount of DRAM installed in the router). At a minimum, each router must meet the system memory requirements for the Cisco IOS software image that it is running.
- The IPv4 connection between the ISP and the customer should be of a high quality. Use the **tracert** and **ping** utilities to determine the endpoint, number of hops, and the round-trip delay of the existing IPv4 connection to the customer. If the IPv4 connection is poor, the IPv6 connection will be poor (because IPv6 traffic will be tunneled over the existing IPv4 connection). Reconnecting the customer through another POP may help remedy a poor connection.
- The available bandwidth on the links that connect to the ISP and customer border routers needs to accommodate the amount of data that each customer needs to route. If the links on the border routers cannot accommodate the data for each customer, choose different border routers or limit the amount of data from each customer.
- Ensure that the ISP Domain Name System (DNS) supports IPv6 AAAA record types in the name-to-address and address-to-name lookup processes. For example, ensure that the DNS is running (or has the equivalent capabilities of) Berkeley Internet Name Domain (BIND) version 9, which provides an implementation of the major components of the DNS for IPv6. DNS configuration is beyond the scope of this document.

- Select interior and exterior routing protocols appropriate for the ISP network configuration. For exterior routing when using IPv6 manually configured tunnels, IPv6 for Cisco IOS software supports multiprotocol BGP extensions for IPv6 and static routes; Routing Information Protocol (RIP) for IPv6 and Integrated Intermediate System-to-Intermediate System (IS-IS) for IPv6 are supported for interior routing. Open Shortest Path First (OSPF) for IPv6 will be supported in a future release of the Cisco IOS software.



Note If IPv6 IS-IS is used as the Interior Gateway Protocol (IGP), generic routing encapsulation (GRE) must be used as the encapsulation protocol; if IPv6 RIP is used as the IGP, GRE or IPv4 can be used as the encapsulation protocol.

The scenario presented in this document uses RIP for IPv6 for interior routing; multiprotocol BGP extensions for IPv6 and static routes are used for exterior routing.

- Obtain a /32 IPv6 prefix from the applicable regional registry (ARIN). (For example, the 2001:yyyy::/32 prefix).
- Delegate a /40 IPv6 prefix (from the /32 IPv6 prefix assigned to the ISP from the regional registry) to the Seattle and San Francisco POPs. (For example, the 2001:yyyy:0300::/40 and 2001:yyyy:0100::/40 prefixes, respectively.)
- Delegate a /48 IPv6 prefix (from the /40 IPv6 prefix of the San Francisco POP) to the customer site that is requesting IPv6 services. (For example, the 2001:yyyy:0101::/48 prefix.)
- Obtain the IPv4 address of the IPv6 exchange point border router that the ISP will use to access the IPv6 Internet. (For example, the 192.168.5.1 address.)
- If the ISP is running Cisco IOS NetFlow applications within its IPv4 infrastructure, ensure that the applications are configured to use port 64 to gather billing, planning, and monitoring statistics on IPv6 traffic (use port 41 to monitor the IPv4 address of an IPv6 tunnel).

Implementation Process Steps

This section explains the following configuration tasks:

- [Configuring an IPv6-Native Connection Between the Seattle and San Francisco POPs](#)
- [Configuring a Tunnel Between the Seattle POP and the IPv6 Exchange Point](#)
- [Configuring a Tunnel Between the San Francisco POP and the IPv6 Customer Site](#)

Configuring an IPv6-Native Connection Between the Seattle and San Francisco POPs

Using the 2001:yyyy:0300::/40 and 2001:yyyy:0100::/40 IPv6 prefixes that were delegated to the Seattle and San Francisco POPs, respectively, configure the identified border router in both POPs to be dual-stacked and configure a dedicated, IPv6-native connection between both border routers. Additionally, enable an IPv6 RIP process globally on both routers and on the router interfaces that will be used to create the IPv6-native connection.



Note BGP routes are redistributed into the IPv6 RIP routing processes on both border routers and a prefix list is configured to filter inbound routing updates on Fast Ethernet interface 1/0.

Seattle POP Border Router

```

ipv6 unicast-routing

interface fastethernet1/0
  description connection to San Francisco POP
  ip address 192.168.1.1 255.255.255.0
  ipv6 address 2001:yyyy:0300:0201::1/64
  ipv6 rip cisco enable

ipv6 router rip seattle-pop
  distribute-list prefix-list list3 in fastethernet 1/0
  default-information originate
  redistribute bgp

ipv6 prefix-list list3 seq 10 deny ::/0
ipv6 prefix-list list3 seq 15 permit ::/0 le 128

```

San Francisco POP Border Router

```

ipv6 unicast-routing

interface fastethernet1/0
  description connection to Seattle POP
  ip address 192.168.1.2 255.255.255.0
  ipv6 address 2001:yyyy:0300:0201::2/64
  ipv6 rip cisco enable

ipv6 router rip sf-pop
  distribute-list prefix-list list3 in fastethernet 1/0
  default-information originate
  redistribute bgp

ipv6 prefix-list list3 seq 10 deny ::/0
ipv6 prefix-list list3 seq 15 permit ::/0 le 128

```

Configuring a Tunnel Between the Seattle POP and the IPv6 Exchange Point

Using the /32 IPv6 prefix from the IPv6 exchange point, manually configure an IPv6 tunnel with IPv4 as both the encapsulation and transport protocol between the border router in the Seattle POP and the border router in the IPv6 exchange point in Chicago. Additionally, configure a multiprotocol BGP peering relationship between the two border routers so that they can exchange IPv6 route information.

Seattle POP Border Router

```

interface serial2/0
  description interface for tunnel to IPv6 exchange point
  bandwidth 10000
  ip address 192.168.4.1 255.255.255.0
  no ip route-cache
  no ip mroute-cache
  load-interval 30

interface Tunnel1
  description tunnel to IPv6 exchange point (over Serial2/0)
  no ip address
  ipv6 address 2001:yyyy:0300:0202::1/64
  tunnel source serial2/0
  tunnel destination 192.168.5.1 255.255.255.0
  tunnel mode ipv6ip

```

```

router bgp 65001
  no synchronization
  no bgp default ipv4-unicast
  bgp log-neighbor-changes
  redistribute rip
  neighbor 2001:yyyy:0300:0202::2 remote-as 65000

  address-family ipv6
    neighbor 2001:yyyy:0300:0202::2 activate
    neighbor 2001:yyyy:0300:0202::2 override-capability-neg
    network 2001:yyyy::/32
  exit-address-family

ipv6 route 2001:yyyy:0300:0201::/64 fastethernet1/0
ipv6 route ::/0 tunnel 1

```

**Note**

Multiprotocol BGP is used to exchange routes with the IPv6 exchange point. The static routes in the example ensure that all traffic destined for network 2001:yyyy:0300:0201::/64 is routed over Fast Ethernet interface 1/0 to the San Francisco POP (not shown in the example) and all other traffic is routed over tunnel interface 1 to the IPv6 exchange point.

IPv6 Exchange Border Router

```

interface serial2/0
  description interface for tunnel to Seattle POP
  bandwidth 10000
  ip address 192.168.5.1 255.255.255.0
  no ip route-cache
  no ip mroute-cache
  load-interval 30

interface Tunnell
  description tunnel to Seattle POP (over Serial2/0)
  no ip address
  ipv6 address 2001:yyyy:0300:0202::2/64
  tunnel source serial2/0
  tunnel destination 192.168.4.1 255.255.255.0
  tunnel mode ipv6ip

router bgp 65000
  no synchronization
  no bgp default ipv4-unicast
  bgp log-neighbor-changes
  redistribute rip
  neighbor 2001:yyyy:0300:0202::1 remote-as 65001

  address-family ipv6
    neighbor 2001:yyyy:0300:0202::1 activate
    neighbor 2001:yyyy:0300:0202::1 override-capability-neg
    network 2001:yyyy::/32
  exit-address-family

ipv6 route 2001:yyyy::/32 Tunnell

```

**Note**

Multiprotocol BGP is used to exchange routes with the Seattle POP. The static route in the example ensures that all traffic destined for network 2001:yyyy::/32 is routed over tunnel interface 1 to the Seattle POP.

Configuring a Tunnel Between the San Francisco POP and the IPv6 Customer Site

Using the 2001:yyyy:0100::/40 IPv6 prefix that was delegated to the San Francisco POP, manually configure an IPv6 tunnel with a static route between the border router in the San Francisco POP and the border router of the customer site in Hawaii that is requesting IPv6 services.

San Francisco POP Border Router

```
interface fastethernet1/1
  description interface for tunnel to IPv6 customer site
  ip address 192.168.2.2 255.255.255.0

interface Tunnel1
  description tunnel to IPv6 customer site (over fastethernet1/1)
  no ip address
  ipv6 address 2001:yyyy:0100:0301::1/64
  tunnel source fastethernet1/1
  tunnel destination 192.168.3.1 255.255.255.0
  tunnel mode ipv6ip

ipv6 route 2001:yyyy:0101::/48 Tunnel1
```



Note

The static route in the example ensures that all traffic destined for network 2001:yyyy:0101::/48 is routed over tunnel interface 1 to the IPv6 customer site.

IPv6 Customer Site Border Router

```
interface serial2/0
  description interface for tunnel to San Francisco POP
  bandwidth 10000
  ip address 192.168.3.1 255.255.255.0
  no ip route-cache
  no ip mroute-cache
  load-interval 30

interface Tunnel1
  description tunnel to San Francisco POP (over Serial2/0)
  no ip address
  ipv6 address 2001:yyyy:0100:0301::2/64
  tunnel source serial2/0
  tunnel destination 192.168.2.2 255.255.255.0
  tunnel mode ipv6ip

ipv6 route 2001:yyyy:0101:0001::/64 fastethernet 1/0
ipv6 route 2001:yyyy:0101:0002::/64 fastethernet 1/1
ipv6 route ::/0 tunnel 1
```



Note

The static routes in the example ensure that all traffic destined for the local networks 2001:yyyy:0101:0001::/64 and 2001:yyyy:0101:0002::/64 is routed over Fast Ethernet interfaces 1/0 and 1/1, respectively, and that all other traffic is routed over tunnel interface 1 to the San Francisco POP.

Device Characteristics and Annotated Configuration Files

Table 1 lists the characteristics of the Seattle and San Francisco POP border routers shown in Figure 3.



Note

For clarity, Table 1 lists only the IP addresses of the Seattle and San Francisco POP border router interfaces that are used to connect to the IPv6 exchange point and the customer site border routers (through the deployment of manual IPv6 tunnels and a native IPv6 connection). The IP addresses of POP border router interfaces that are not used to connect to the exchange point and customer site are not listed in the table.

Table 1 POP Border Router Hardware and Software Characteristics

Router Characteristic	Seattle POP Border Router	San Francisco POP Border Router
Host name	seattle-pop	sf-pop
Chassis type	Cisco 7206VXR	Cisco 7206VXR
Physical interfaces	2 serial (T3) 2 Fast Ethernet	2 serial (T3) 2 Fast Ethernet
Software loaded	Cisco IOS Release 12.2(8)T IP Standard feature set Software image: c7200-p-mz	Cisco IOS Release 12.2(8)T IP Standard feature set Software image: c7200-p-mz
Memory	16 MB Flash 128 MB DRAM	16 MB Flash 128 MB DRAM
Addresses	Fast Ethernet 1/0: 2001:yyyy:0300:0201::2/64 serial 2/0: 192.168.4.1/24 Tunnel 1: 2001:yyyy:0300:0202::1/64	Fast Ethernet 1/0: 2001:yyyy:0300:0201::1/64 Fast Ethernet 1/1: 192.168.2.2/24 Tunnel 1: 2001:yyyy:0100:0301::1/64

Following are the annotated running configuration files for the Seattle and San Francisco POP border routers shown in Figure 3.



Note

Display text that is not pertinent to the IPv6 network topology shown in Figure 3 was removed from the following configuration files.

Seattle POP Border Router

```
! Identify the version of Cisco IOS software running on the router
!
version 12.2
!
! Include timestamps on log and debug entries that are useful for
! troubleshooting and optimizing the network.
!
service timestamps debug datetime localtime show-timezone
service timestamps log datetime localtime show-timezone
!
! Specify that passwords will be encrypted in configuration output.
!
```

```

service password-encryption
!
! Configure the router name
!
hostname seattle-pop
!
! Configure boot options
!
boot system flash slot0:
boot system flash bootflash:
!
! Configure logging
!
logging buffered 10000 debugging
!
! Configure secret password
!
enable secret 5 [removed]
!
! Configure clock timezone and summertime rule
!
clock timezone PST -8
clock summer-time PDT recurring
!
!
ip subnet-zero
no ip source-route
no ip rcmd domain-lookup
!
! Configure router domain name
!
ip domain-name ISPDomain.com
!
! Configure DNS name servers
!
ip name-server 192.168.1.10
ip name-server 192.168.2.21
ip name-server 2001:yyyy:0300:0201::5
!
! Enable IPv6 routing
!
ipv6 unicast-routing
!
! Configure the physical and tunnel interfaces
!
interface fastethernet1/0
description connection to San Francisco POP
ip address 192.168.1.1 255.255.255.0
ipv6 address 2001:yyyy:0300:0201::1/64
ipv6 rip cisco enable
!
interface serial2/0
description interface for tunnel to IPv6 exchange point
bandwidth 10000
ip address 192.168.4.1 255.255.255.0
no ip route-cache
no ip mroute-cache
load-interval 30
!

```

```

interface Tunnell
  description tunnel to IPv6 exchange point (over serial 2/0)
  no ip address
  ipv6 address 2001:yyyy:0300:0202::1/64
  tunnel source serial2/0
  tunnel destination 192.168.5.1 255.255.255.0
  tunnel mode ipv6ip
!
! Configure IPv6 routing protocols and neighbor peerings
!
ipv6 router rip seattle-pop
  distribute-list prefix-list list3 in fastethernet 1/0
  default-information originate
  redistribute bgp
!
router bgp 65001
  no synchronization
  no bgp default ipv4-unicast
  bgp log-neighbor-changes
  redistribute rip
  neighbor 2001:yyyy:0300:0202::2 remote-as 65000
!
  address-family ipv6
    neighbor 2001:yyyy:0300:0202::2 activate
    neighbor 2001:yyyy:0300:0202::2 override-capability-neg
    network 2001:yyyy::/32
    exit-address-family
!
! Configure basic IP routing
!
ip default-gateway 192.168.4.1
ip classless
ip route 0.0.0.0 0.0.0.0 192.168.4.1
!
! Configure IPv6 prefix lists
ipv6 prefix-list list3 seq 10 deny ::/0
ipv6 prefix-list list3 seq 15 permit ::/0 le 128
!
! Configure IPv6 static routes
!
ipv6 route 2001:yyyy:0300:0201::/64 fastethernet1/0
ipv6 route ::/0 tunnel 1
!
end

```

San Francisco POP Border Router

```

! Identify the version of Cisco IOS software running on the router
!
version 12.2
!
! Include timestamps on log and debug entries that are useful for
! troubleshooting and optimizing the network.
!
service timestamps debug datetime localtime show-timezone
service timestamps log datetime localtime show-timezone
!
! Specify that passwords will be encrypted in configuration output.
!
service password-encryption
!
! Configure the router name
!
hostname sf-pop

```



```

!
! Configure boot options
!
boot system flash slot0:
boot system flash bootflash:
!
! Configure logging
!
logging buffered 10000 debugging
!
! Configure secret password
!
enable secret 5 [removed]
!
! Configure clock timezone and summertime rule
!
clock timezone PST -8
clock summer-time PDT recurring
!
!
ip subnet-zero
no ip source-route
no ip rcmd domain-lookup
!
! Configure router domain name
!
ip domain-name ISPDomain.com
!
! Configure DNS name servers
!
ip name-server 192.168.1.10
ip name-server 192.168.2.21
ip name-server 2001:yyyy:0100:0301::5
!
! Enable IPv6 routing
!
ipv6 unicast-routing
!
! Configure the physical and tunnel interfaces
!
interface fastethernet1/0
description connection to Seattle POP
ip address 192.168.1.2 255.255.255.0
ip address 2001:yyyy:0300:0201::2/64
ipv6 rip cisco enable
!
interface fastethernet1/1
description interface for tunnel to IPv6 customer site
ip address 192.168.2.2 255.255.255.0
!
interface Tunnel1
description tunnel to IPv6 customer site (over Serial2/0)
no ip address
ipv6 address 2001:yyyy:0100:0301::1/64
tunnel source fastethernet1/1
tunnel destination 192.168.3.1 255.255.255.0
tunnel mode ipv6ip
!
! Configure IPv6 routing protocols
!
ipv6 router rip sf-pop
distribute-list prefix-list list3 in fastethernet 1/0
default-information originate
redistribute bgp

```

```
!  
! Configure basic IP routing  
!  
ip default-gateway 192.168.4.1  
ip classless  
ip route 0.0.0.0 0.0.0.0 192.168.4.1  
!  
! Configure IPv6 prefix lists  
!  
ipv6 prefix-list list3 seq 10 deny ::/0  
ipv6 prefix-list list3 seq 15 permit ::/0 le 128  
!  
! Configure IPv6 static routes  
!  
ipv6 route 2001:yyyy:0101::/48 Tunnel1  
!  
end
```

Related Documents

Refer to the following documents for additional information about IPv6 for Cisco IOS software:

- *IPv6 Deployment Strategies*
- *IPv6: Connecting to the 6bone Using Manually Configured Tunnels*
- *IPv6: Connecting to the 6bone Using 6to4 Tunnels*
- *Interconnecting IPv6 Domains Using Tunnels*
- *Start Here: Cisco IOS Software Release Specifics for IPv6 Features*
- *Implementing IPv6 for Cisco IOS Software*
- *IPv6 for Cisco IOS Software Command Reference*

Refer to the following website on Cisco.com for more information on the Cisco implementation of IPv6:

<http://www.cisco.com/ipv6>