IPv6 Networking Fundamentals

Release 1.1

January 2012

Technical Assistance

This is an AT&T proprietary document developed for use by AT&T customers. For additional technical assistance contact your AT&T sales team. (This document was prepared by the AT&T Network Design and Consulting Division.)

Legal Disclaimer

This document does not constitute a contract between AT&T and a customer and may be withdrawn or changed by AT&T at any time without notice. Any contractual relationship between AT&T and a customer is contingent upon AT&T and a customer entering into a written agreement signed by authorized representatives of both parties and which sets forth the applicable prices, terms and conditions relating to specified AT&T products and services, and/or, to the extent required by law, AT&T filing a tariff with federal and/or state regulatory agencies and such tariff becoming effective. Such contract and/or tariff, as applicable, will be the sole agreement between the parties and will supersede all prior agreements, proposals, representations, statements or understandings, whether written or oral, between the parties relating to the subject matter of such contract and/or tariff.

Table of Contents

0 Release Updates, 1.0 to 1.1

- Updated all IPv6 addresses to the Industry standard documentation address
- Added details about client DHCPv6 process
- Added an use case for NAT64 for inbound server access

1 Introduction to IPv6

This document provides an introduction to IPv6 networking, discusses methods for configuring networks, and highlights major differences from IPv4. A key concept to understand is that IPv4 and IPv6 have different packet header formats and, as such, should be considered to operate as completely independent networks. An IPv4 packet can never be delivered directly to an IPv6 interface protocol stack, nor can an IPv6 packet be delivered to an IPv4 stack. Packet data payloads wanting to cross between networks would need to be "translated" between formats. As networks migrate from IPv4 to IPv6 it should be expected that translation services and dual network ("dual-stack"[1]) operations will occur for extended periods. This document will discuss IPv6 networking, addressing, and transition methods from IPv4 to IPv6.

1.1 Why IPv6?

The popularity of the Internet has driven us to the brink of exhaustion of IPv4 addressable space. We have been hearing for a number of years that the industry is running out of IPv4 address space. The situation becomes more urgent every year and yet companies still fully operate on IPv4 and continue to add endpoints and new services. This is possible because companies innovate new ways to be frugal with the addresses that remain in IPv4. The main conservation tool has been "Private Addressing" (as defined in RFC1918), where chunks of the available IPv4 space are reserved for private use only. This allows large private networks to hide behind a few public addresses and use Network Address Translation (NAT) to sponsor access to the public Internet.

NAT has developed significant momentum among network administrators for its isolation qualities and, in some sense, has become a security "best practice." That very isolation quality of NAT is what many IPv6 proponents say stifles the end-to-end connectivity required for creative new applications and innovative network usage models. For example, cellular wireless endpoints are emerging as promising recipients of creative new networking and application models. As cellular radios and GPS receivers get smaller and cheaper to deploy, the notion of tracking everything and anything (packages, rail cars, vehicles, appliances, etc) will place an increased burden on the need for uniquely addressable and easily reachable endpoints. Couple this with the fact that all voice traffic will soon be carried as IP data requiring unique endpoint identification, and the requirement for IPv6-sized address space starts to look inevitable and imminent. IPv6 contains 3.4 x 10^{38} addressable endpoints. Numbers this large are hard to grasp but one might think of it as much larger than all the grains of sand on the earth, or all the stars in all the galaxies. In comparison, IPv4 is capable of supporting 4 billion addresses at best with the actual number being less due to losses associated with sub netting.

1.2 IPv6 Notable Features

In addition to much larger address space, IPv6 provides several improvements over the existing IPv4 protocol. Some of these improvements are discussed below.

IPSec--One of the key IPv6 enhancements is the integration of IPSec. Authentication Header (AH) and Encapsulation Security Payload (ESP) have been included as optional IPv6 extension headers and no longer treated as an upper layer protocol. IPSec on IPv6 operates the same way as IPv4 IPSec in

[1] For information on AT&T's Dual-Stack AVPN service, see "AT&T VPN Dual-Stack (IPv4 / IPv6) Configuration Guide."

providing secure communication between endpoints. One key difference is IPv6 provides the capability to build a secure link between any IPv6 endpoints without the use of intermediate VPN gateways.

Stateless Auto-configuration--IPv6 introduces a new method of address assignment called Stateless Auto-configuration. This allows an IPv6 endpoint to dynamically assign itself an IP address from route advertisements that are announced on the connected link. Stateless Auto-configuration is enabled by default on all operating systems running the IPv6 protocol stack. Therefore, users do not need to manually or dynamically (via DHCP) assign an IPv6 address to endpoints as is commonly done on IPv4 networks. Due to its current limitations, network administrators may want to use Stateless Auto-configuration in combination with DHCPv6 or manual address configuration to provide other network information such as DNS addresses. This will be discussed in greater detail in Section 3.

Flow-based QoS—The Flow Label field was added to the IPv6 header to allow routers to identify and provide faster QoS treatment of packets as they are forwarded through an IPv6 network. In traditional IPv4 networks, routers must read into layer-3 and layer-4 headers to classify the specific flow of traffic. A flow is typically defined as a combination of source/destination address, transport protocol, and port number. Once the flow is matched, IPv4 routers map the appropriate QoS marking in the layer-3 Type of Service field. With the introduction of Flow Label, IPv6 routers will be able to classify traffic without reading the layer-4 header and thus provide faster QoS treatment of packets. Please see Section 2.2, Figure 2.2, for a comparison of IPv4 and IPv6 headers.

Router Efficiency—Two notable IPv4 header fields were not included in IPv6. They are “Fragmentation Offset” and “Header Checksum.” In IPv4 networks, routers are responsible for fragmenting packets if they are larger than the Maximum Transmission Unit (MTU) of an intermediate transport link. Fragmentation Offset is used to identify each fragmented packet which allows receivers to piece the fragmented packets back to its original form. Routers must allocate additional memory and CPU cycles to temporarily store and process each fragment. In IPv6, however, packet fragmentation is now the responsibility of IPv6 endpoints. ICMPv6 messages are used to discover the Path MTU (PMTU) between the source and destination. Based on the ICMPv6 response, IPv6 endpoints adjust packet sizes appropriately to avoid sending packets larger than the lowest MTU supported on the path. In theory, IPv6 endpoints should never send packets that are larger than the network can handle. However it is important to note ICMPv6 messages will only get responses from other IPv6 elements. Therefore if IPv4 tunnels are used within the path, ICMPv6 will not be aware of the supported MTU of the underlying network. In this case, some packets may be dropped by the underlying tunnel network. To avoid this issue, network administrators must ensure the correct MTU is defined on the IPv6 tunnel interfaces.

Today, routers inspect each IPv4 packet to ensure the IP header has not been altered. This function is used to quickly identify and discard malformed packets from being processed further by the router, and was originally introduced to address potentially poor transmission quality of the intermediate transport links. With advances in technology it has become less of an issue and end-to-end checks are performed at a higher layer anyway. Therefore, “Header Checksum” was left out of IPv6 relieving routers from the burden of running checksum calculations and comparison.

Multicast—In IPv4 networks, multicast applications are limited to the corporate WAN since they are not supported on the IPv4 Internet. There are no globally routable IPv4 multicast addresses. IPv6, on the other hand, has allocated globally routable multicast addresses that could be used over the Internet. This allows companies to apply for multicast address spaces similar to IPv6 global addresses. Internet service providers (ISPs) may choose not to support multicast in their initial offer. As IPv6 becomes more popular and new multicast enabled applications are created, ISPs may be pressured by customers and competitors to support multicast.

1.3 NAT and Security

With IPv6, there is no longer a need to use private IP addresses in the LAN with such a large pool of IPv6 addresses available. In fact, it is expected for customers to assign public IPv6 addresses to LAN/WAN devices and to avoid NAT. This may be a difficult concept for many network and security administrators to accept. Many believe NAT provides a level of security by hiding internal devices and networks from the outside world. But private IP addressing does not provide comprehensive security. Perimeter security is provided through solid security rules and policies that are applied to firewalls and edge devices to prohibit foreign traffic from entering their networks. Similar policies can be applied to IPv6 traffic.

In some ways IPv6 can be more secure from potential attacks. For instance, with IPv4 most connected locations use subnets ranging form /24 to /16, which can translate to endpoint addresses numbering between 256 (2^8) and 65536 (2^{16}). In attempts to unlawfully access a network, hackers often execute network scans sweeping for open TCP/UDP ports and searching for vulnerabilities. With IPv6, hackers must scan much larger blocks of addresses due to the minimum atomic subnet size of /64. Service providers may assign a network prefix allowing /48 or /56 subnetting space per location (see Section 2 for details of IPv6 addressing). Even with /56 allocations hackers are faced with a remaining 2^{72} addresses to scan. Most network firewalls will easily block ping sweeps of addresses. So let's assume hackers will utilize deep TCP/UDP scanning methods probing 65536 ports per IP address. This translates to (2)*(2^{72})*(65536) probes to completely scan all ports and addresses of /56 subnet. Let's also assume a hacker has a 100Mbps Internet connection and decides to deep scan a /56 subnet issuing a 100-byte packet per port. It will take more than $1.6*10^{14}$ years to complete the scan from start to finish. In comparison, it would take under 5 minutes to perform the same scan of /24 IPv4 subnet addresses and ports. Therefore it will be pointless to use these traditional methods to probe IPv6 networks.

1.4 Who is Moving to IPv6?

The United States government attempted to impose the early adoption of IPv6 based on its anticipation of IPv4 address exhaustion and its interest in the benefits provided by IPv6. The Office of Management and Budget (OMB) mandated that all agencies support IPv6 by June 2008. The deadline has passed, and many government agencies have yet to fully embrace IPv6. However on September 28, 2010, the White House Chief Information Officer released a memorandum setting a deadline for the federal agencies to implement IPv6 by the end of FY2012 (for the Internet) and the end of FY2014 (for the private WAN). Many private enterprises have similarly delayed their migration plans. The adoption of IPv6 has been slow for several possible reasons:

- Use of IPv4 conservation methods such as NAT.

- Lack of accessible resources on IPv6 networks providing no incentive for vendors to invest in developing IPv6 applications or to justify spending capital that generates little revenue in the near-term.

- Limited support for IPv6 among equipment vendors (firewalls, reporting and analysis tools, etc)

On February 2011, IANA allocated the last /8 IPv4 to address block to APNIC. This news created renewed interest in IPv6 by many enterprises leading up to the IPv6 World Day on June 8, 2011. In fact, the number of IPv6 Internet routes nearly doubled during this period. Most participating enterprises used IPv6 World Day to test their Internet facing servers to ensure proper operation and connectivity over the IPv6 Internet. Others simply used this as an opportunity to market themselves as IPv6 ready organization. Whatever the reasons may be, IPv6 is unavoidable. There is a limited pool of available IPv4 addresses.

Each day, this pool is getting smaller. Regional Internet Registrars have made IPv4 address application processes more difficult and are strongly encouraging their customers towards IPv6. In the next two to three years, most ISPs will run out of IPv4 addresses and will be forced to distribute IPv6-only addresses to their customers. How will these customers access your business if you do not have an IPv6 presence?

1.5 Potential Deployment Challenges

Deploying an IPv6 network is not trivial and requires careful planning. IPv6 should not be dismissed as just another layer-3 protocol. Customers should pay close attention to IP address and allocation methods, IPv4/IPv6 feature differences, and Service Provider policies. For instance, Cisco routers do not presently support Virtual Router Redundancy Protocol (VRRP) for IPv6. If you have a multi-vendor environment, you may have VRRP running in your network. What solution do you implement in place of VRRP?

Another example is firewalls. Most companies block ICMP packets to prevent hackers from probing their networks. With IPv6, ICMP is used instead of “ARP” messages to resolve MAC/IPv6 addresses and to discover MTU size across a network via Path MTU ICMP queries. Therefore, customers will need to make allowance for these packets where they are not required in today’s IPv4 networks.

Customers will also have to tackle some important addressing questions. Should I use AT&T provided IPv6 addresses or should I obtain my own Provider Independent addresses? Can I assign and advertise IPv6 addresses obtained through ARIN over European or Asian Internet providers? Answers to these questions depend on your requirements as well as the Service Providers policies. This document discusses several issues that customers may encounter.

2 IPv6 Address

This section briefly describes the elements of IPv6 that should be understood by anyone expecting to participate in an IP network architecture at any level.

2.1 Defining and Recognizing IPv6 Addresses

The industry has become sufficiently comfortable with the decimal expression of 4 octets that are commonly used to represent IPv4 addresses. If someone says "192-dot-168-dot-1-dot 1" we immediately know they are describing a private address typically used by consumer broadband (cable or DSL) routers. IPv6 addresses are represented by 8 double octets expressed in hex format which can be listed in three primary forms.

1. Fully Expanded Form expresses an IPv6 address in its entirety with each hex digit displayed.

2. Common Expanded Form shortens each double octet to express only its value (i.e. "1" instead of "0001"),

3. Compressed Form allows for the replacement of consecutive sets of zeroed octets with a double colon (::). The double colon can obviously be used only once in each address representation.

The differences are represented in figure 2.1.1.

IPv4: 12.106.96.3 – Familiar format of 4 decimal octets

IPv6: X:X:X:X:X:X:X:X – Uses 8 double octets in Hex form

Common Expanded Form

2001:DB8:20:0:0:0:0:1

Fully Expanded Form

2001:DB8:0020:0000:0000:0000:0000:0001

Compressed Form

http://[2001:DB8:20::1]:5800

Try to remember this address:

{ 2001: 0DB8:0020:a1ef:d01c:effa:cad6:a19e }

Figure 2.1.1—IPv6 Address Format

Also note in Figure 2.1.1's "Compressed Form" how port number specification requires additional brackets to distinguish between uses of the colon. The last example in Figure 2.1.1 shows how cryptic addresses commonly become. As shown, it will be very difficult to remember these addresses and may lead to possible human errors if these addresses are manually assigned.

Figure 2.1.2 shows how addresses are broken up hierarchically for assignment to different entities ranging from regional registries to end users. Each regional registry has already been assigned a /23 subnet by the Internet Assigned Numbers Authority (IANA). Service Providers (SP) such as AT&T have been or will be assigned /32 subnets from the regional registry. Those SPs who have a global presence are required to obtain IPv6 blocks from each regional registry where they operate. As a general rule, IPv6 addresses obtained in one region are not allowed to be advertised into another region. There are exceptions, but as a rule, IPv6 address blocks obtained in North America through one SP are not allowed to be advertised in Europe over another SP's network. It must be aggregated to the SP's /32 subnet and can not be advertised by its smaller allocated subnet.

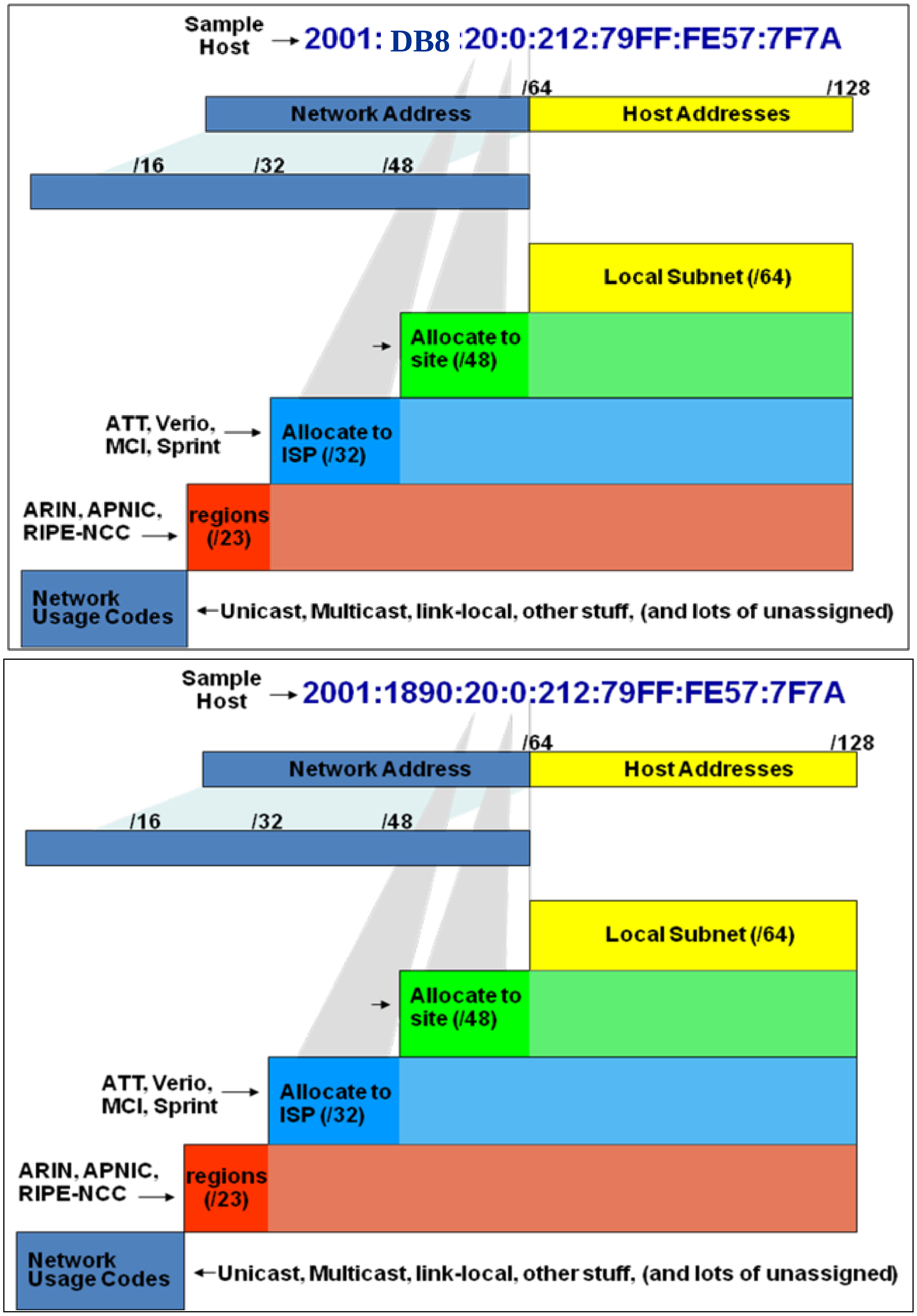


Figure 2.1.2—IPv6 Address Hierarchy

This policy is primarily to control the amount of BGP routes that are advertised and maintained by an SP's routers. As it is today, the IPv4 Internet has in excess of 400,000 routes that are actively maintained by Internet routers (2011) compared to about 75,000 in the year 2000. That is roughly about 22,000 new routes each year. With IPv6, hierarchical addressing and routing are encouraged with the expectation that companies will obtain IPv6 addresses from their SPs. However there are some major issues that stand in the way; namely IPv6 Internet multi-homing[2]. Since SP-allocated IPv6 address blocks are not allowed to be advertised on another SP's network, how will customers be able to failover to the surviving Internet service using SP-allocated IPv6 addresses? At this point, there isn't a clear solution addressing this issue other than requesting Provider-Independent (PI) IPv6 addresses from regional registries. Please check with your SP, but PI addresses should be allowed on any SP's network without regional boundaries. PI addresses will enable customers to advertise the same address block across different SP's networks.

Those customers without multi-homing requirements can obtain IPv6 addresses directly from their SPs. Based on individual addressing requirements, each customer will likely be allocated subnet space from /64 to /48 from the SPs. Larger customers may be allocated bigger subnets if sufficient justification is provided. Customers can further divide the allocated block on their internal networks. The smallest assignable subnet is expected to be the /64 atomic subnet, primarily to support Stateless Auto-Configuration[3].

Figure 2.1.2 shows how the final 64 bits are used to define the host address. Host addresses will typically be assigned in a /64 space allotment which allows for a very large, flat subnet if needed. This convention has been adopted to accommodate the auto-configuration of every host. Depending on the host operating system, it may use the MAC address of the host following the EUI-64 standards. This further allows the unique identification of hardware devices wherever they happen to connect to the network since their MAC address can be derived by their source IP address. This convention makes it difficult for anyone to actually remember their IP address. Some are concerned and cite privacy issues with uniquely identifying the hardware within the IP address. Under this convention your laptop can be uniquely identified as the source of any session wherever it connects and whatever it does. Therefore the concept of "temporary" addresses has been introduced to eliminate the reliance on EUI-64 based addresses and to address privacy concerns. The "temporary" addresses are generated using an algorithm that randomly generates the last 64-bits of the IPv6 address.

Some resist the auto-configuration model on convenience grounds preferring to use DHCP to hand out addresses that are more recognizable (1, 2, 3, etc, rather than 212:79FF:FE57:7F7A). Other difficulties with the new format include the likelihood for typographical errors in the specification of static routes when configuring routers. One bright spot might be the definition of the default route. Typically defined by "0.0.0.0 0.0.0.0" in IPv4, IPv6 can simply use "::/0" for the default route. Another benefit of the /64 host address space is the infeasibility of ping sweeps and port scans over such a huge space thus concealing nodes from an external attack as discussed previously in Section 1.3.

2.2 New IP Layer Header

Specific differences between IPv4 and IPv6 can be seen in the difference between the IP Header layouts as shown in Figure 2.2. The primary difference is the significant increase in the size of the address field. Those fields alone make the IP header of an IPv6 packet much larger. This could be an important factor for applications which use smaller packets, such as voice-over-IP.

[2] Multi-homing refers to the practice of connecting to two or more service providers from a single location usually for reliability purposes.

[3] Stateless Auto-configuration, and more specifically EUI-64, is a technique that uses the MAC address of the hardware interface to fully define the host address. It is expected to be in common use and presumes to use the entire lower /64 of address bits. This effectively defines /64 as the smallest assignable subnet for compatibility across the industry, but does not preclude special case assignment of smaller subnets.

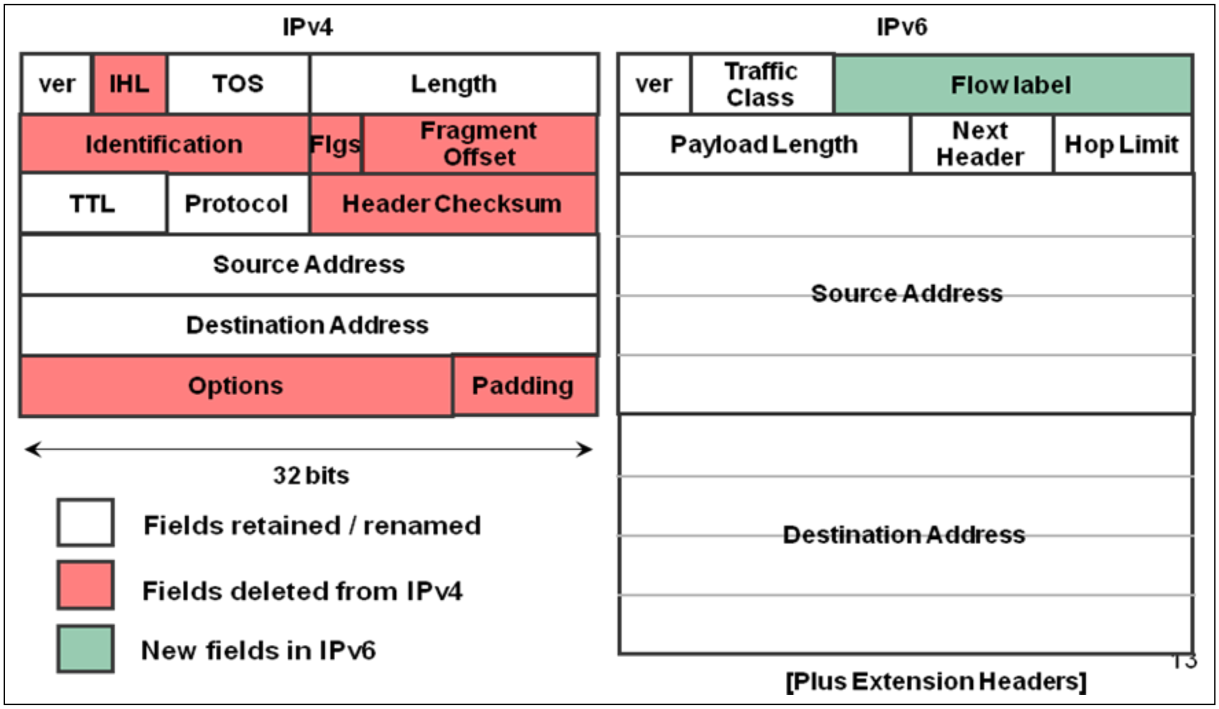


Figure 2.2—IPv4 / IPv6 Header Fields

Several fields in the IPv4 header have been removed resulting in a more streamlined IPv6 header as mentioned in Section 1.3. Additional capabilities (such as end-to-end IPSec) are handled by extension headers. The “Flow Label” field was also added to IPv6 headers to allow for more specific classification of layer-3 traffic flows. This potentially allows routers to take faster action on IPv6 packets compared to IPv4 packets where routers might need to access information from various places in order to define specific traffic flows (i.e. IP address, Port, and Transport Protocol).

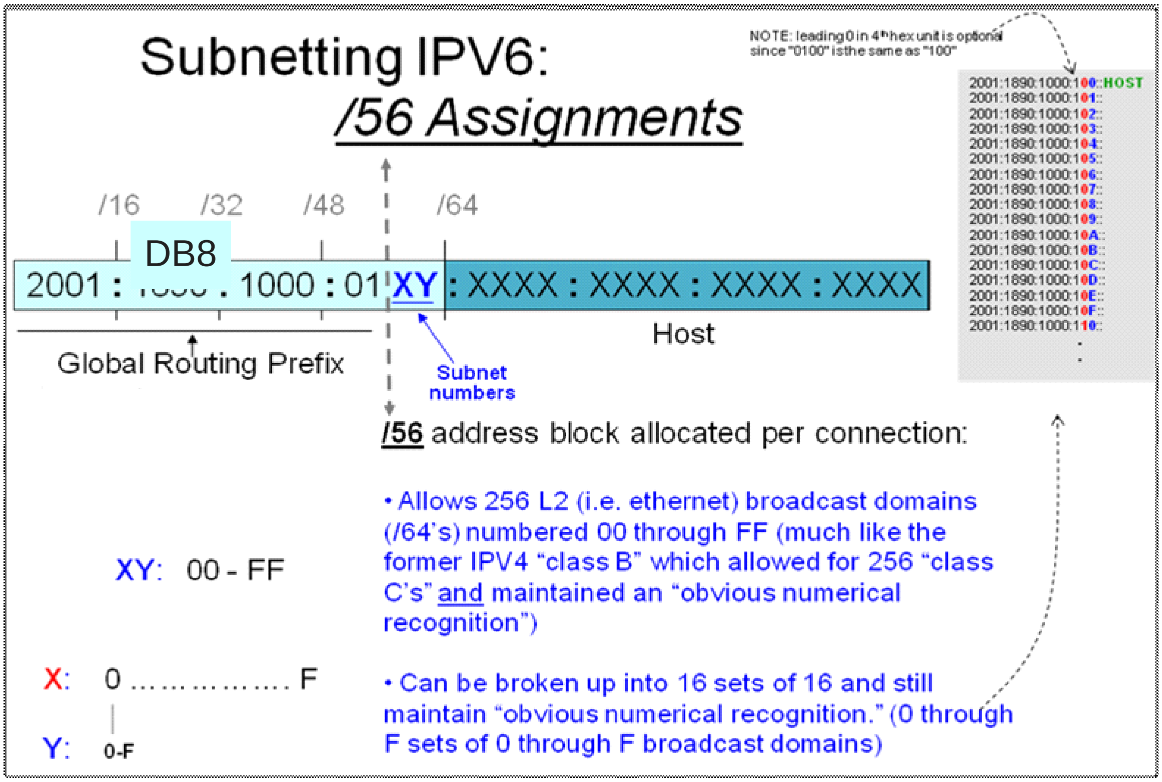
Even though many fields have been removed, the four fold increase in the IPv6 address field still results in an IP header that is nearly twice the size that it used to be. With small packets and real time services like VoIP expected to proliferate, the increased header size adds to the packet and processing overhead. If mobile-IP, IPSec, and other services defined in header extensions become a requirement, the overhead can get much worse. In addition, an IPv6 header can contain an infinite number of extension headers, which not only further increases overhead, but may also provide a new target for hackers.

2.3 New Addressing Format

In the IPv4 world, network administrators typically developed favorite methods for defining networks and subnets. Before address depletion became an issue, a favorite format was to use an IPv4 “/16” subnet at each location and “/24’s” within that “/16” for subnets at that location. This allowed for easy recognition of the addressing subdivisions since they exactly followed the octet separation. This convention is still often followed in the private address space.

In the public space, however, address depletion has forced the use of smaller subnets that don’t necessarily align with easily recognizable boundaries. Recognizable subnets will hopefully make a comeback with IPv6. For example, the AT&T AVPN service currently proposes to assign “/56” sized

IPv6 subnets to each customer location. Figure 2.3 illustrates how this allows for 256 of the "/64" atomic subnets with easy identification in the address field (00 through FF in the XY position shown). It also allows further breakdown into 16 sets of 16 sets, etc. It is unlikely subnetting would need to be smaller than is definable by the octets. Note that using hex digits allows the use of half octets (4 bits) while still maintaining recognizable numbering of the subnets.



Subnetting IPV6:
/56 Assignments

NOTE: leading 0 in 4th hex unit is optional since "0100" is the same as "100"

```
2001:1890:1000:100::HOST
2001:1890:1000:101::
2001:1890:1000:102::
2001:1890:1000:103::
2001:1890:1000:104::
2001:1890:1000:105::
2001:1890:1000:106::
2001:1890:1000:107::
2001:1890:1000:108::
2001:1890:1000:109::
2001:1890:1000:10A::
2001:1890:1000:10B::
2001:1890:1000:10C::
2001:1890:1000:10D::
2001:1890:1000:10E::
2001:1890:1000:10F::
2001:1890:1000:110::
.
.
```

/56 address block allocated per connection:

- Allows 256 L2 (i.e. ethernet) broadcast domains (/64's) numbered 00 through FF (much like the former IPV4 "class B" which allowed for 256 "class C's" and maintained an "obvious numerical recognition")
- Can be broken up into 16 sets of 16 and still maintain "obvious numerical recognition." (0 through F sets of 0 through F broadcast domains)

Figure 2.3—Subnetting IPv6

It remains to be seen how network administrators will approach subnetting in IPv6. It is possible the significant increase in size and complexity of numerical address representations will shift emphasis away from routinely using these numbers in favor of DNS names.

2.4 Address Scope

While it is widely expected that IPv6 networks will primarily use global addresses in the corporate LAN/WAN, it is important to highlight other address types that are part of IPv6. These include: "node-local," "link-local," "unique-local," and "global" addresses. Unlike IPv4, an IPv6 endpoint can assume multiple IP addresses on a single network interface. A single interface can be assigned global, unique-local, and link-local addresses simultaneously. In fact, a single interface can be assigned multiple global addresses but only one link-local address. In practice, an interface may have multiple global addresses possibly for multi-homed networks. Other address types listed below may be used for internal purposes to establish a private network or to develop applications which only reference internal processes.

- Node-local is not a routable address. It’s defined as ::1/128 and is assigned to all IPv6 endpoint. It is equivalent to the IPv4 127.0.0.1 loopback address and serves no major purpose in network routing. It is typically used by software to pass data to itself.

- A link-local address is required on all IPv6 endpoints. It plays a key role in the Address Resolution Process (ARP). IPv6 replaces IPv4 ARP with Neighbor Solicitation and Neighbor Advertisement messages to resolve the underlying layer-2 MAC address for a destination IPv6 address. The “FE80::” prefix has been allocated for link-local addresses. A link-local address can be used to communicate between IPv6 endpoints (on a shared link) without the need of a global or unique-local address. However, it is not routable and can be used only within a shared link.

- Unique-local addresses are similar to RFC1918 private addresses. They can be deployed within an organization or across multiple organizations but are not routable over the Internet. Customers should read RFC 4193 and follow the process to generate a prefix to reduce the chance of overlapping addresses.

Customers are expected to use global addresses over unique-local addresses across their WAN and LAN. These addresses are globally unique and thus are routable on the Internet. Customers can obtain these addresses from their service providers or, with proper justification, directly from their regional registries like ARIN in North America. Global prefixes obtained directly from the Internet registries are considered Provider-Independent (PI). A PI prefix should be allowed to be advertised across any provider or region regardless of which region the prefix was obtained. Conversely, a global prefix assigned by a service provider does not have this flexibility. Service provider prefixes will likely be tied to the provider and the region, which means the assigned prefix can not be advertised over another provider’s network. It will also be restricted to the region where it was allocated. Therefore those customers with dual-carrier requirements should request PI addresses.

3 IPv6 Address Allocation

IPv6 provides several enhancements to allocating and assigning IP addresses to endpoints. With IPv4 there are two well-understood methods: manual (or static) and Dynamic Host Configuration Protocol (DHCP). These methods are commonly used in IPv4 and are well understood. With the introduction of IPv6, there is now an option called stateless auto-configuration which enables IPv6-enabled endpoints to automatically assign themselves IPv6 addresses based on the prefix advertised over the connected link. In addition, DHCP has been overhauled to support a new service that allows the DHCP server to allocate an entire block or subnet to an endpoint. These enhancements were developed to help in addressing and readdressing of IPv6 endpoints as users change service providers or prefixes. This section discusses these address allocation options and identifies their benefits and pitfalls for users to consider as they deploy IPv6 into their network.

3.1 Stateless Auto-Configuration

Stateless auto-configuration is a method of assigning IPv6 addresses to endpoints defined by RFC 2462. It is enabled by default on interfaces that support the IPv6 protocol stack and requires no additional configuration on IPv6 endpoints. IPv6 endpoints dynamically learn 64-bit IPv6 prefixes from layer-3 IPv6-enabled network devices on the shared link and append its host address to complete the 128-bit IPv6 address as shown below in Figure 3.1.

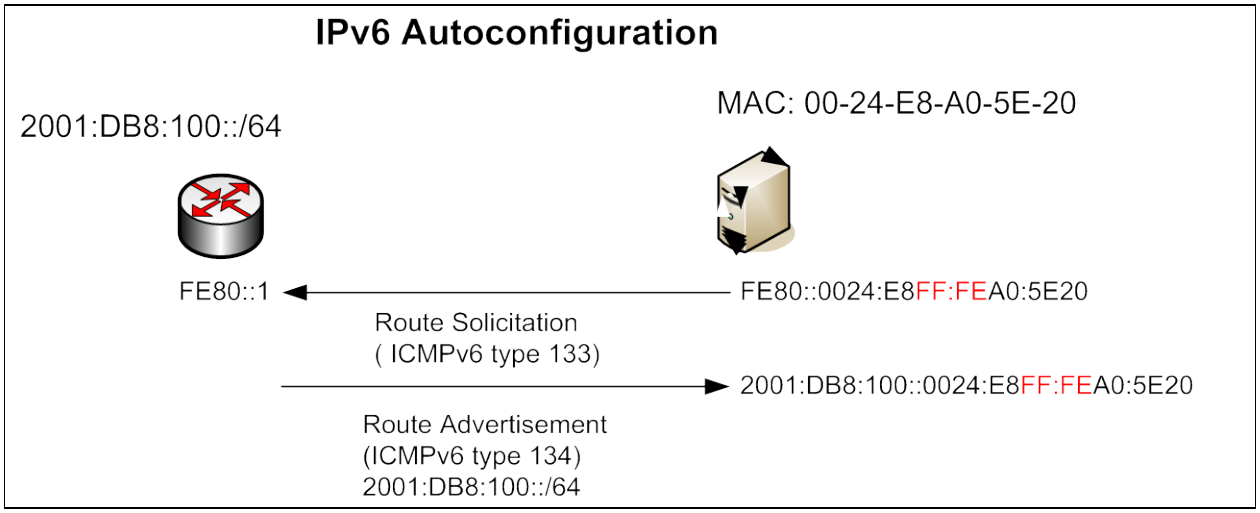


Figure 3.1—IPv6 Stateless Auto-configuration (EUI-64)

Auto-configuration is accomplished through an ICMPv6 message called “Route Advertisement” (RA). By default, Cisco routers will send RA messages every 200 seconds on each IPv6-enabled interface. RA contains an IPv6 prefix or prefixes that are configured on the Ethernet interface. Other information is passed along with each RA such as a prefix lifetime timer that defines how long an endpoint should maintain the prefix in its routing table.

Upon receiving the RA, some endpoints will automatically assign themselves an IPv6 address using the EUI-64 standard which uses the MAC address to build the 64-bit host address. For instance, the 48-bit

MAC address is broken up into two 24-bit blocks, and “FFFE” hex digits are appended between the two 24-bit blocks to complete the 64-bit host address. Some endpoints may use the RFC 4941 by randomly generating the last 64 bits of the address with no dependency on the MAC address. Once an address has been assigned, each endpoint will take an additional measure to ensure no duplicate addresses are present on the link by using Duplicate Address Detection or DAD messages. If the EUI-64 standard is used, there should be no duplicate addresses on the link since MAC addresses are unique. This can provide unique identification of the host hardware within the IP address. However privacy advocates have raised concerns over the use of the EUI-64 standard because it reveals identities of its users based on the MAC address.

Therefore a new addressing standard has been defined by RFC 4941 that advocates the use of random host addresses in lieu of EUI-64. Depending on the operating system, customers may see EUI-64 addresses, random addresses, or both on an IPv6 endpoint. For instance, Windows XP still uses EUI-64 addresses while Windows 7 uses random addresses by default. Customers must issue a **netsh**[4] command to enable the use of EUI-64 addresses on Windows 7.

There may be situations where there are multiple layer-3 devices on the shared link advertising the same prefix. In such a situation, endpoints can determine the default gateway based on the priority setting on the gateways. There are three priority settings on Cisco routers--high, medium, and low. This feature is available on release 12.4(2)T or later. By default, RAs are sent with medium priority. Configuring the RA announcements with high priority will influence the endpoints to use it as the default gateway when there are multiple layer-3 devices advertising different prefixes. An IPv6 endpoint will assign itself multiple IPv6 addresses on a single interface -one for each prefix it learns.

This presents an interesting issue regarding the selection of a default route. Which layer-3 device will the endpoint choose as the default gateway? With no support for dynamic routing protocol, IPv6 endpoints will typically add a static default route for each layer-3 default gateway that advertises the RA message. Based on the priority setting discussed above, a default route will be injected into the endpoint with a lower metric to make a router the preferred default gateway. In addition, endpoints may need to be configured with additional IPv6 routes to influence the traffic to take the correct outgoing path other than the default path; otherwise, endpoints will use RFC3484 (method for default address selection in IPv6) to determine the best default gateway for the destination.

There may also be cases where there are no layer-3 devices to advertise an IPv6 prefix over a shared link. In this situation, IPv6 endpoints will attempt to solicit layer-3 devices to obtain a prefix. However if there is no prefix announced, then the endpoint will only assign itself a link-local address which is non-routable. By default, a link-local address is assigned to all IPv6-enabled interfaces in addition to routable IPv6 addresses. Link-local address can be used to communicate between other IPv6 endpoints on the shared link but can not be used across an enterprise since it is not a routable address.

Auto-configuration does simplify the task of assigning IPv6 addresses. An entire LAN could be renumbered by simply changing the IPv6 prefix on the default gateway. Through the RA process, endpoints will learn the new IPv6 prefix of the gateway and assign themselves addresses from the new prefix.

While auto-configuration simplifies some things, it presents other challenges for network administrators. For one, it does not pass the DNS address information in its RA messages. DNS must be manually configured or used in conjunction with stateless DHCPv6. (Note: If endpoints are dual-stacked, IPv4 DNS servers can be used to resolve domain names to IPv6 addresses.) Auto-configuration also presents potential network and security issues. For instance, a router can be mis-configured with a bad IPv6 prefix which will advertise false information on its connected link. IPv6 endpoints will receive the prefix and

[4] Netsh is a Microsoft Windows command.

then update their interface with an additional IPv6 address. This bad prefix may be used as the default gateway effectively black-holing traffic. This same vulnerability can be exploited by hackers to advertise false RAs on the link and wreak havoc on the network.

In addition, customers should pay close attention to the prefix and RA timers to achieve the desired results. By default, Cisco routers set prefix timers at 30-days and default RA timers at 30 minutes, which means IPv6 endpoints can potentially maintain static routes and addresses for the prefix for 30-days if left alone. A default route, on the other hand, is pruned from the PC's routing table relatively fast because of Neighbor Unreachability Detection. If a network is readdressed, then there is a good chance the previous IPv6 address will remain on the endpoint until the lifetime timer expires (30-days).

3.2 DHCPv6

DHCPv6 provides the capability to automatically allocate IPv6 addresses and other configuration parameters to IPv6 endpoints. There are two types of DHCPv6 services: stateful and stateless. Stateful DHCPv6 enables the server to pass full configuration parameters including IPv6 addresses, DNS, refresh/lifetime timers, etc, while stateless DHCPv6 provides configuration information like DNS and NTP[5] server addresses without assigning or maintaining an IPv6 address to the endpoint. Stateless DHCPv6 does not require update messages to maintain a stateful session with the server, which results in reduced network traffic. Stateless DHCPv6 would likely be used in conjunction with stateless auto-configuration to provide complete network information to IPv6 endpoints.

Similar to IPv4 DHCP service, DHCPv6 supports server, client, and relay modes. In addition, DHCPv6 supports prefix delegation which allows a DHCPv6 server to allocate an entire IPv6 prefix block to an endpoint which can be reassigned to other endpoints. In this mode, a network device is configured as a DHCPv6 client on the interface toward the central DHCPv6 server, and as a server on the interface towards the DHCPv6 endpoints on the LAN as shown in Figure 3.2.1.

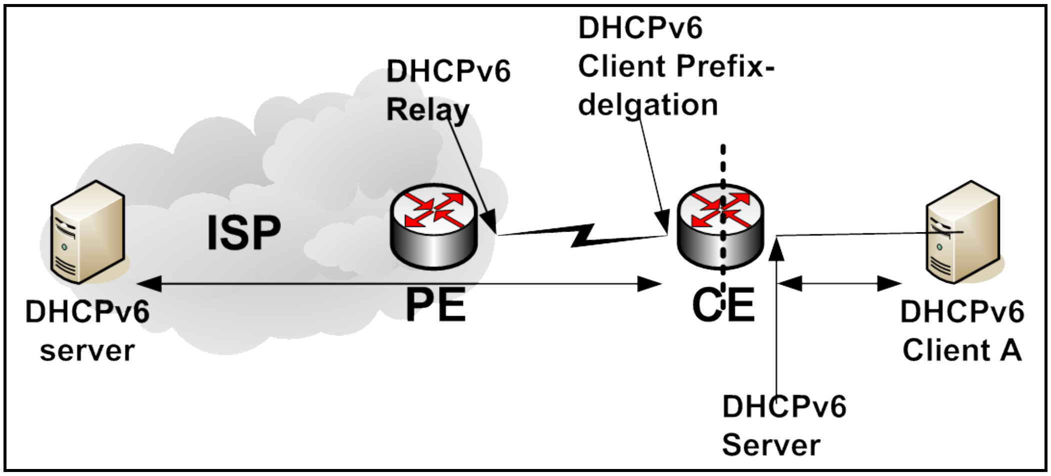


Figure 3.2.1—DHCPv6 Prefix-delegation

[5] NTP refers to network time protocol which is typically used to synchronize events.

Please note for Cisco routers, a router interface can only be configured in one DHCPv6 mode. For instance, an interface can not be configured in client and relay modes at the same time. Therefore the DHCPv6 Client A in Figure 3.2.1 will receive its IPv6 address from the CE router and not from the DHCPv6 server in the ISP's network. Figure 3.2.1 implies service provider support for network prefix delegation. This may not be the case. Customers intending to deploy this method and assuming service provider support would first need to check with their service provider.

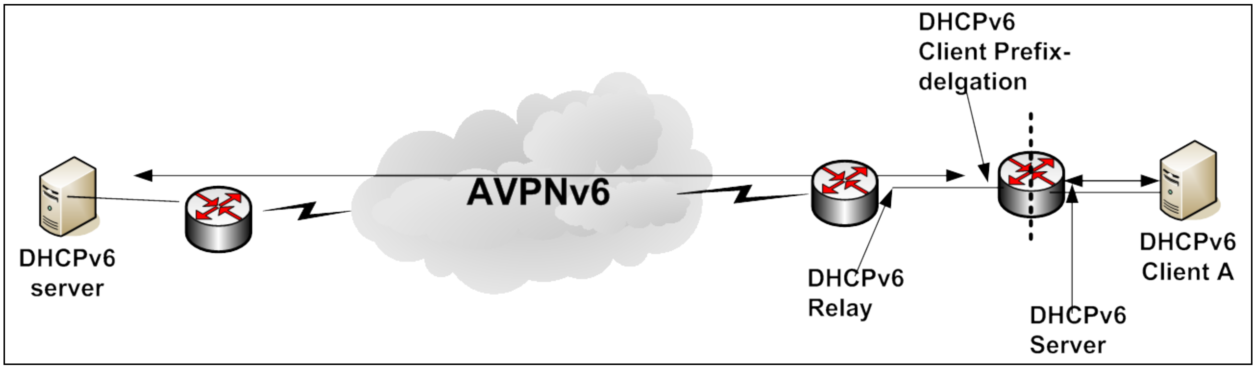


Figure 3.2.2—DHCPv6 Prefix-delegation w/o ISP Relay

As noted, the scenario depicted in Figure 3.2.1 relies on the service provider to enable the DHCPv6 relay service on the link connected to the CE router. Without it, DHCP requests are dropped by the PE router. If this is the case, customers can implement a design similar to Figure 3.2.2, where the CE router is configured as the DHCPv6 relay. This allows the CE router to forward the DHCP request directly to the global address of the DHCPv6 server. However, customers must incorporate a solution to advertise the delegated prefix out of the CE router toward another Layer-3 device (router) configured both as a DHCP endpoint and server. As shown in Figure 3.2.2, this intermediate router whose interface is facing the CE router must be configured as the DHCP client while the inside interface towards the end client is configured as a DHCP server.

It is important to note that some operating systems may not support DHCPv6 service. For instance, Windows XP only supports Stateless Auto-configuration with no support for DHCPv6. In this case, Windows XP endpoints may need to use 3rd party DHCP software like Dibbler[6]. Windows 7 on the other hand does include support for DHCPv6 service, but it can be somewhat confusing to configure. For instance, DHCP for IPv4 is enabled by accessing the "Properties" page and checking the box to "Obtain IP address automatically". However, checking the "Obtain IPv6 address automatically" in Windows 7 does not necessarily enable DHCPv6 service on the client. If the Windows 7 client is on a standalone network with no IPv6 network gateways configured, then the client would be by default operating in DHCPv6 client mode. If it is not in client mode, There are two primary commands that are used in combination to enable stateless and stateful DHCPv6 processes:

- **netsh interface ipv6 set interface <InterfaceNAME> managedaddress=enabled** --enables the stateful DHCPv6 process.

- **netsh interface ipv6 set interface <InterfaceNAME> otherstateful=enabled** --instructs the endpoint to request and obtain other DHCPv6 information such as DNS.

[6] Dibbler is a freeware DHCPv6 software— http://klub.com.pl/dhcpv6

When the **otherstateful** option is enabled without **managedaddress**, the system is configured as a stateless DHCPv6 endpoint. Conversely, if the **otherstateful** option is disabled and the **managedaddress** option is enabled, the IPv6 endpoint will receive its IPv6 address dynamically but will not receive other stateful information. However, these configuration statements are subordinate to the DHCP options embedded in the RA message, and the client will assume the DHCPv6 role defined in the RA messages. By default, RA messages do not have the **managedaddress** and the **otherstateful** bits enabled. This means the IPv6 client will not be in the DHCPv6 client mode. Therefore it would be the best practice to configure the IPv6 default gateways with the **managedaddress** and **otherstateful** options to force the clients into DHCPv6 client mode. . This effectively instructs all DHCPv6 enabled endpoints to change from Auto-configuration to a stateless or stateful DHCP process. This approach eliminates the need to configure each client as described previously. The Cisco command snippets to enable DHCPv6 options are:

ipv6 nd managed-config-flag **enables endpoints for stateful DHCPv6**

ipv6 nd other-config-flag **enables endpoints to accept other DHCPv6 information**

3.3 Manual

Manual configuration provides users full control over statically assigning IPv6 addresses to endpoints. This is not suitable for large networks because they are especially prone to errors in light of the fact that an IPv6 address is 128 bits long and uses the less familiar hexadecimal. Manual configuration is likely to be used on devices that require static addresses like network routers, switches, and servers.

4 IPv4/IPv6 Transition Solutions

4.1 Overview

This section provides details of several transition mechanisms that can be deployed to provide connectivity to IPv6 resources over an IPv4 WAN. These transition options will likely be utilized in situations where there are islands of IPv4 and IPv6 networks within a corporate network during a migration to IPv6, as most companies will not have the luxury to convert their entire corporate network to IPv6 all at once.

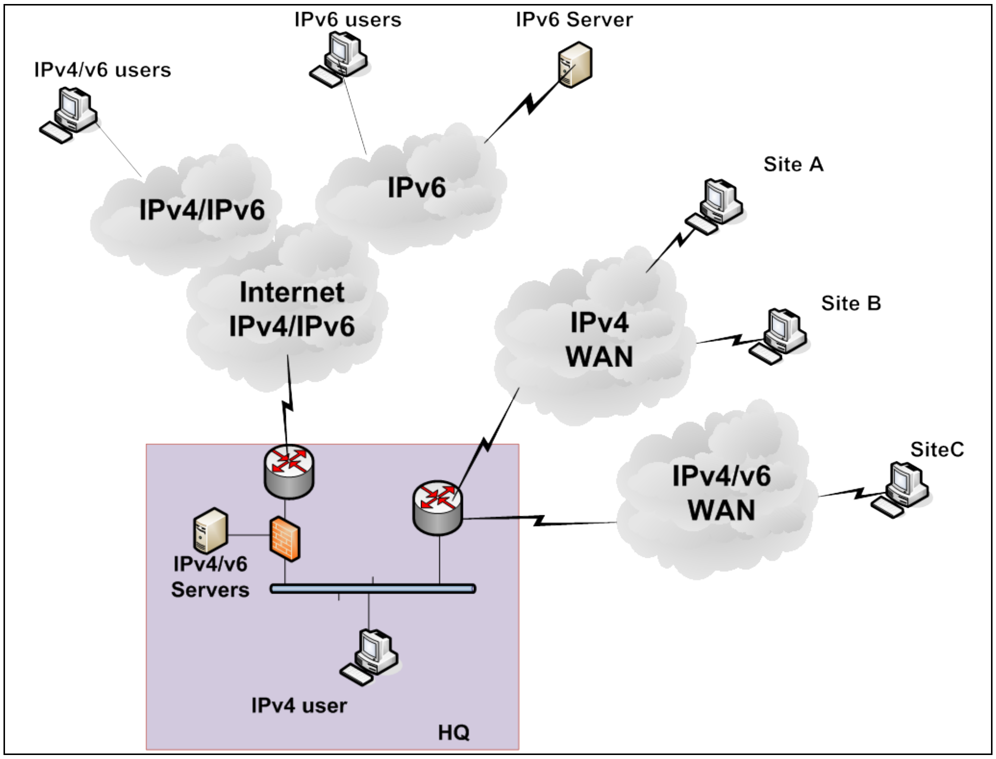


Figure 4.1—Network in transition to dual-stack environment

Figure 4.1 represents a typical network in transition to a dual-stack LAN and WAN. Most companies will eventually migrate their Internet service to dual-stack to make their Internet servers accessible to IPv4 and IPv6 Internet users. As next steps, companies will convert their corporate networks to dual-stack. As depicted in Figure 4.1, Site C has converted its LAN and WAN to support dual-stack while Site A and B remain on IPv4 WAN. Site C has access to both IPv4 and IPv6 resources. However, Site A and B can only access IPv4 resources. Site C will be able to communicate with Site A and B via their IPv4 address. A challenge is presented when these IPv4-only endpoints require access to IPv6 applications, servers, or Internet. How can IPv6 packets be transported over this IPv4 WAN to access IPv6 resources at the Hub or the Internet? This section will shed some light on solutions to address this challenge. Table 4.1 represents six transition options that will be discussed.

Table 4.1 – Transition Options

Transition Options	Feature Description
Generic Routing Encapsulation (GRE)	Point to point tunneling protocol that enables IPv6 packets to be transported inside IPv4 packets. Typically deployed on WAN CE routers over an IPv4 WAN.
6to4 Tunneling	Similar to GRE except it provides point to multipoint tunnels and simplifies Hub router configuration
Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)	6to4 adapter tunneling protocol supported on IPv6 clients to build IPv4 tunnels to ISATAP servers to transport IPv6 packets.
Teredo Tunnel	6to4 adapter tunneling protocol supported on IPv6 clients that allows IPv4 tunnels to be built through NAT or firewall devices
Network Address Translation (NAT)	NAT is not expected to be an integral part of IPv6 networks. However some form of NAT may be incorporated in a network to translate IPv6 to IPv6 and IPv6 to IPv4 addresses.
Proxy Server	Dual-stack proxy servers provide access to IPv4 and IPv6 destination resources to IPv4-only or IPv6-only corporate users.

4.2 Generic Routing Encapsulation

Generic Routing Encapsulation (GRE) is a tunneling protocol that is frequently used today to encapsulate a packet inside another IP packet to essentially hide the presence of the original packet from the underlying IPv4 network. GRE enables networks that are protected behind GRE tunnels to communicate directly with each other using their native protocols. In the IPv6 world, GRE can be used to transport IPv6 packets between two IPv6 islands that are connected by an IPv4 network as documented in RFC 2473.

Figure 4.2 illustrates a network that has migrated a site to dual-stack LAN and WAN while some sites remain on IPv4. Site A is completely defined by IPv4 addresses on its LAN as well as its WAN connection to the Hub. Without any change, Site A will only be able to communicate with IPv4 networks. Site B has migrated to dual-stack LAN but has yet to convert its WAN connection to dual-stack or IPv6. In order for Site B to communicate with other IPv6 networks, its Customer Edge (CE) router must be configured with a GRE tunnel to the Hub as shown. Once the tunnel is established between Site B's and the Hub CE's routers, IPv6 packets can be encapsulated inside IPv4 packets and transported over the IPv4 WAN between these two sites.

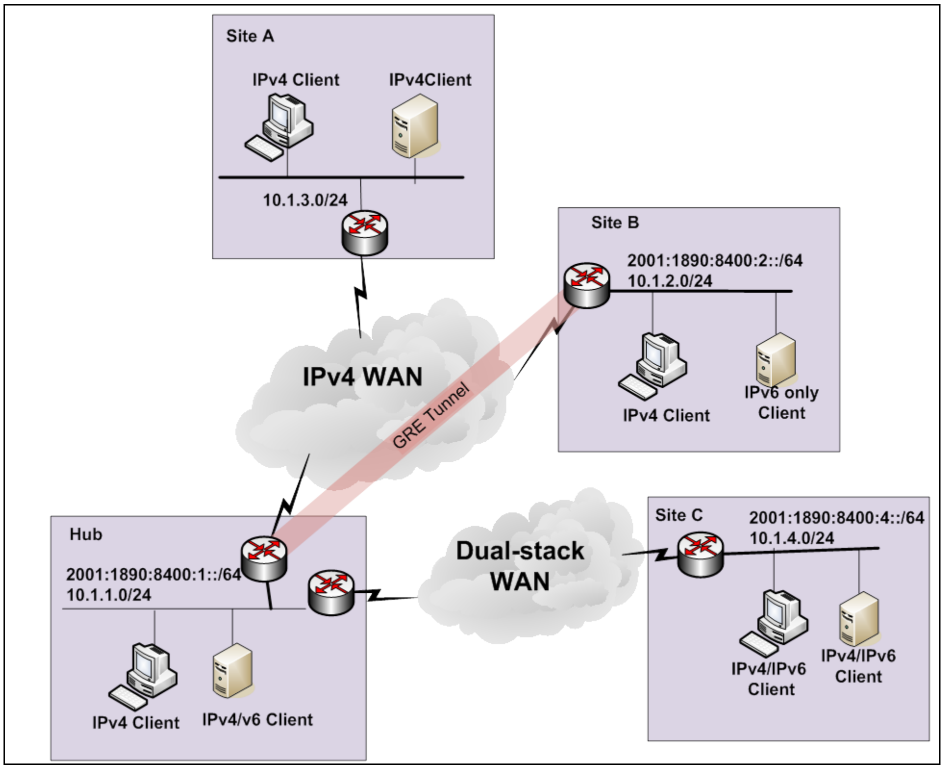


Figure 4.2—IPv6 in IPv4 GRE Tunnel

Companies should be mindful of some disadvantages of using GRE tunnels. First, they add an additional IP overhead of 24 bytes beyond the additional 40 bytes of IPv6 overhead. For each packet that is transmitted, there is a minimum of 64 bytes of overhead. This doesn't include the IPv6 Extension Headers that can be appended to each packet. This can cause an issue with large packets being dropped by the IPv4 WAN network.

As mentioned in Section 1.2, IPv6 uses ICMP messages generated by endpoints to discover the smallest MTU that the path between two communicating endpoints can support. IPv6 network routers and switches no longer perform packet fragmentation. IPv6 networks expect the responsible endpoints to discover the allowable MTUs along the path (PMTU) and adjust the payload accordingly. However in a network incorporating GRE tunnels, ICMP messages are not recognized by the underlying IPv4 WAN since they are encapsulated inside an IPv4 packet. Therefore Path MTU discovery does not take into account the MTU supported by the underlying IPv4 network.

If MTU is incorrectly configured on the GRE tunnel, endpoints can potentially send larger packets than the IPv4 network can handle. For instance, assume the IPv4 network supports a 1500 byte MTU, and the GRE tunnel is set to 1500 MTU. IPv6 endpoints will send IPv6 packet with 1500 bytes. A CE router will append 24 bytes for the IP and GRE header which creates a 1524 byte packet that will be dropped by the IPv4 WAN network since it exceeds what it can support.

4.3 Automatic 6to4 Tunneling[7]

Automatic 6to4 Tunneling (here referred to as 6to4) is a clever way to embed an IPv4 address in an IPv6 address and create a stateless mesh of tunnels. This is very useful in environments where IPv6 segments are isolated from one another due to IPv4 only network elements. As an example, Figure 4.3 illustrates a topology consisting of several IPv6 LANs that are unable to communicate with each other natively across an IPv4 only WAN.

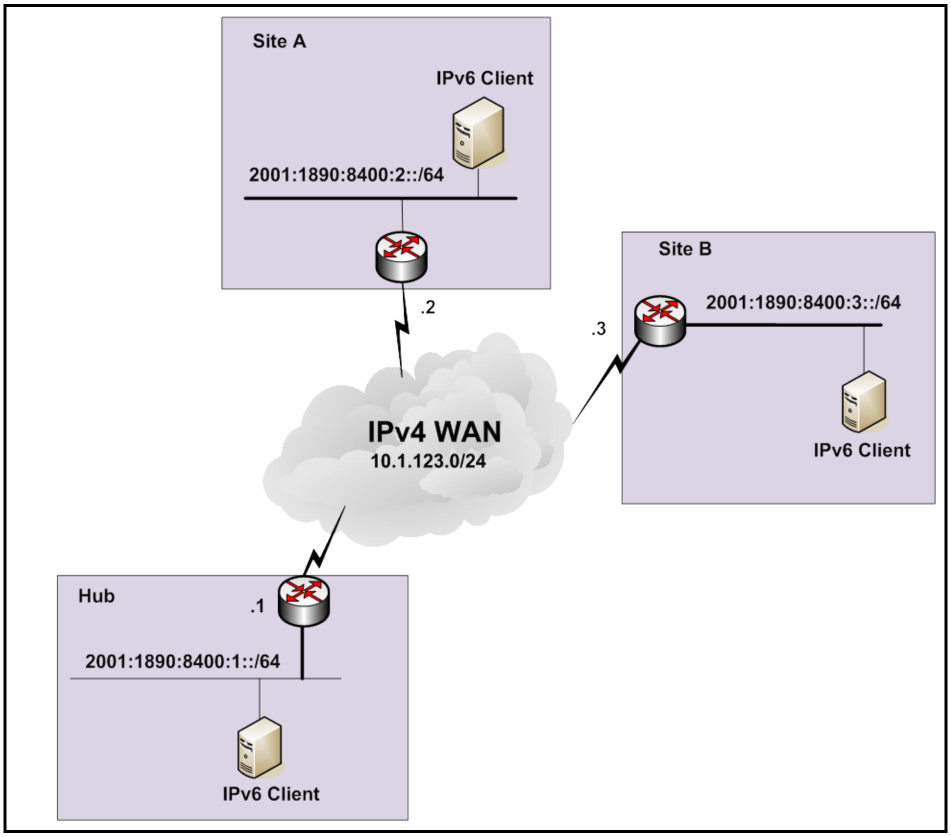


Figure 4.3—6to4 Tunnel

If all three sites need to establish IPv6 communications then a tunneling mechanism is necessary to carry the traffic across the IPv4 WAN. As described in the previous section, GRE tunnels are a viable solution. However, if full mesh connectivity is required, the number of GRE tunnels required quickly becomes unmanageable. This simple example with only 3 sites would of course only require 3 tunnels however a 50 site network would require 1,225 tunnels[8].

Using 6to4 we are able to conceptually route traffic destined for an IPv6 network using an IPv4 next hop. If we were to configure such a route from the example Hub WAN router to Site A IPv6 LAN in a Cisco router, the configuration statement conceptually would look like: **ip route 2001:EEEE:2::/64 10.1.123.2**. This is of course an illegal configuration and will result in an error message but it is the basis of 6to4 tunneling.

[7] RFC3056, “Connection of IPv6 Domains via IPv4 Clouds”, http://www.ietf.org/rfc/rfc3056.txt

[8] Number of Tunnels Required = (NumSites * NumSites-1) / 2

Specifically, 6to4 tunnel addresses use the format **2002:<IPv4 Address in Hex>:<16 bit Network Number>::/64**. By embedding a real IPv4 address that is known and reachable across the WAN, the 6to4 address signals the router to extract the IPv4 next hop and send an IPv4 encapsulated IPv6 packet. Creating a 6to4 multipoint tunnel interface and a static route that points all IPv6 traffic destined for 2002::/16 out the tunnel interface allows communications to be establish between all the isolated IPv6 LANs. As an example of the traffic flow, when the Hub WAN router attempts to send IPv6 traffic to Site A (in Figure 4.3), the router will first look in its routing table and determine that the output interface is a 6to4 tunnel interface (due to a static route to the 2002… next hop). Next, the Hub router will convert the next hop address to an IPv4 address and encapsulate the packet in an IPv4 header. More details are available in RFC 3056.

4.4 Intra-Site Automatic Tunnel Addressing Protocol[9]

Intra-Site Automatic Tunnel Addressing Protocol (ISATAP) is well defined in RFC 5214. ISATAP is an IPv6 tunnel protocol that allows companies to deploy IPv6 capability to individual endpoints. With GRE tunneling, an entire site is essentially provided access to an IPv6 network. With ISATAP, an endpoint is enabled to build an IPv4 tunnel to an ISATAP server. Once the connection has been established with the ISATAP server, endpoints can transmit IPv6 packets over IPv4 networks. ISATAP is supported on most operating systems (OS) including Windows XP, Vista, Windows7, and various Linux OS. In some cases, systems may need to be manually enabled for IPv6 in order to use ISATAP. Windows Vista and Windows 7 have IPv6 enabled by default (of which security administrators should take careful note).

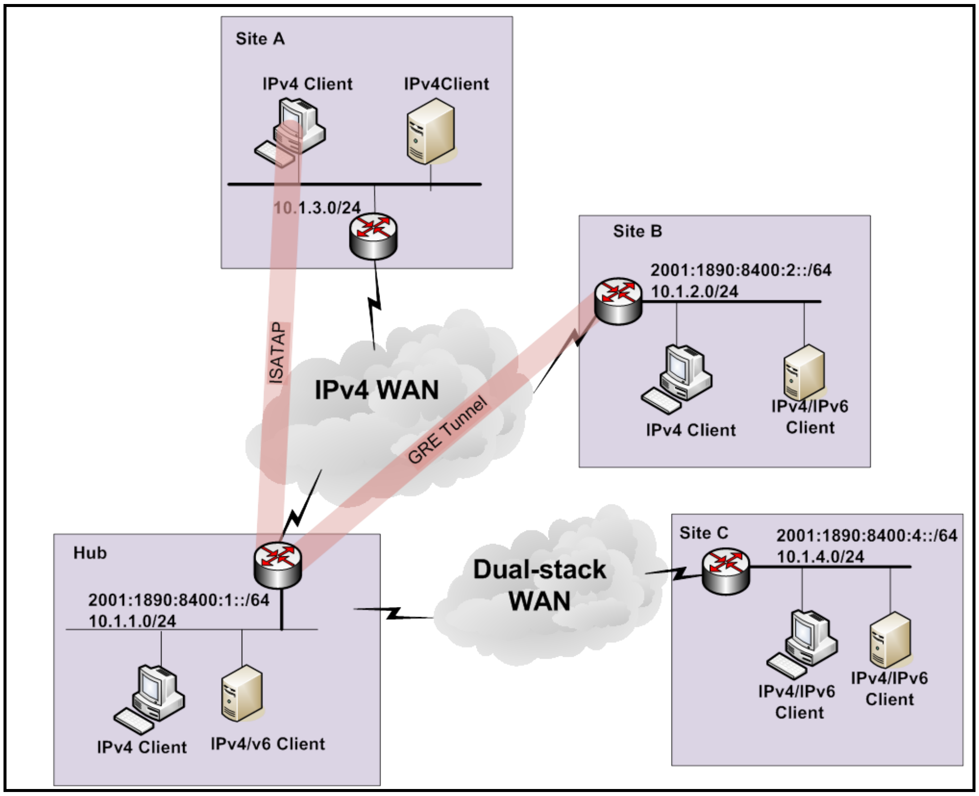


Figure 4.4--ISATAP Solution

[9] RFC5214, "Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)", http://www.ietf.org/rfc/rfc5214.txt

In Figure 4.4, the CE router at the Hub site is used as the corporate ISATAP server. Please note, the CE router is used as an example, and it should not be construed as a best practice design to deploy a router as an ISATAP server. Once the endpoint has been configured for ISATAP, it will attempt to communicate with the ISATAP server via its known IPv4 address. Once the IPv4 communication has been established, the ISATAP server will send a Router Advertisement with its IPv6 prefix to the ISATAP endpoint. The endpoint uses the IPv6 prefix to automatically assign itself an IPv6 address based on EUI-64 standards. ISATAP endpoints can continue to use IPv4 applications, but it is now able to connect to IPv6 resources through the ISATAP tunnel as well. All ISATAP endpoints served by this server will be members of the same IPv6 prefix that has been configured on the server.

For every endpoint that requires IPv6 access, ISATAP must be configured to access the ISATAP server. The ISATAP protocol may be supported on most operating system, but unlike other IPv6 tunneling protocols, ISATAP must be enabled manually or through startup scripts.

An advantage of ISATAP is that neither LAN nor WAN needs to support IPv6. ISATAP was specifically developed to work over IPv4 networks. With the previous GRE solution, the LAN was migrated to dual-stack. This is not required for ISATAP. ISATAP also can be deployed with a redundant IPv6 network through the use of the ISATAP domain name instead of its IP address. Since ISATAP endpoints will continue to use IPv4 DNS servers to resolve domain names, ISATAP endpoints can be configured to use the domain name of the ISATAP server to deploy a more resilient service. If the primary server is not available, endpoints can receive an alternate ISATAP server's IP address to provide access to IPv6 networks.

As an entire site is converted to a dual-stack network, customers must remember to turn off ISATAP services on the endpoint. Otherwise, it will potentially cause IPv6 routing issues since these endpoints may be assigned multiple IPv6 addresses with the IPv6 default route directed through the ISATAP tunnel rather than the newly activated dual-stack interface.

4.5 Teredo Tunnel[10]

Teredo is another tunneling protocol that enables IPv4 endpoints access to IPv6 networks per RFC 4380. It is similar to ISATAP in that it encapsulates IPv6 packets inside IPv4 packets to tunnel through the underlying IPv4 network. However, they differ in their use. ISATAP is designed to be used within a corporate network to bridge IPv6 islands together over IPv4 WAN. Teredo, on the other hand, is designed to provide existing IPv4 users a method to access an IPv6 Internet, potentially through a NAT device. It is seen as more of an Internet solution rather than a corporate WAN solution.

Teredo is supported on most operating systems that support an IPv6 protocol stack. On newer operating systems, particularly Windows Vista and Windows 7, Teredo is enabled by default and connects to the Microsoft Teredo server: teredo.IPv6.microsoft.com. Once the connection has been established with the public Teredo server, it is assigned a public IPv6 address with access to the public IPv6 Internet. Since Teredo is a tunnel technology, IPv6 packets will pass through most firewalls without being inspected. This opens up the Teredo endpoint for potential vulnerabilities unchecked by traditional security devices. While useful for residential users, Teredo is not seen as a practical solution for corporate networks.

[10] RFC4380, "Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs)", http://www.ieft.org/rfc/rfc4380.txt

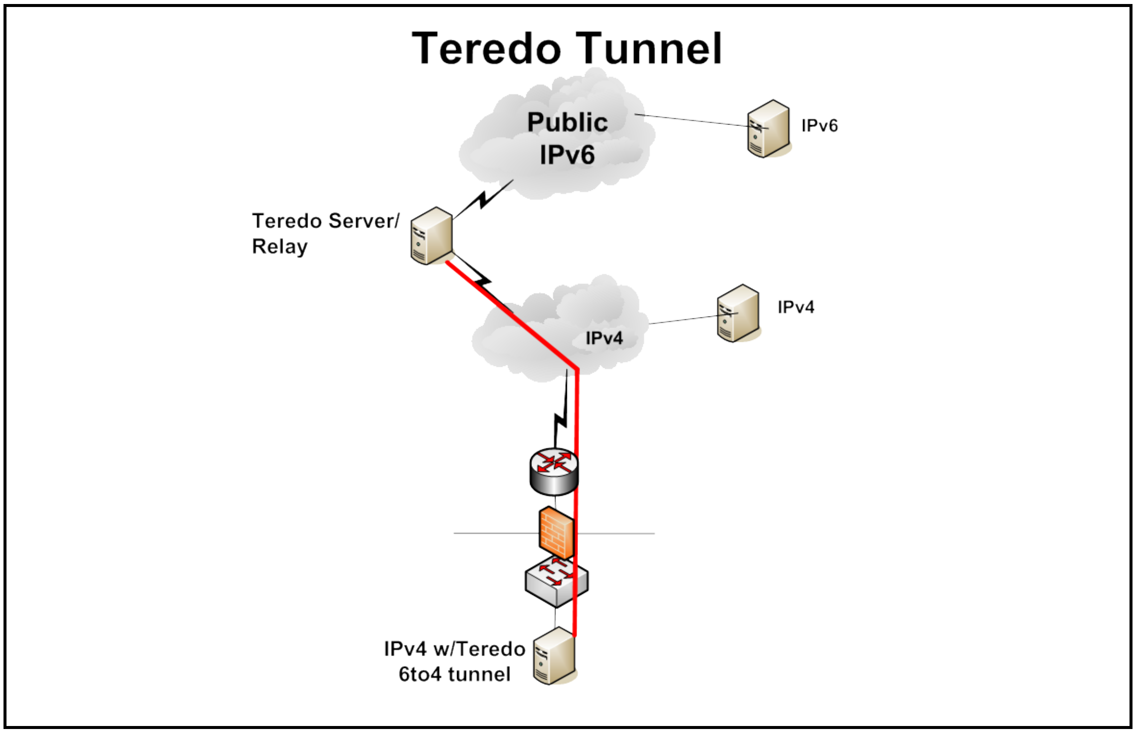


Figure 4.5—Teredo Tunnel

There are several well known public Teredo servers:

- teredo.remlab.net / teredo-debian.remlab.net (France)
- teredo.autotrans.consulintel.com (Spain)
- teredo.IPv6.microsoft.com (USA, Redmond) (default for WindowsXP/2003/Vista/2008 OS)
- teredo.ngix.ne.kr (South Korea)

These servers are available to all users who are interested in accessing the IPv6 Internet. Since they are public servers, they may not be as reliable due to capacity, utilization, and availability.

4.6 Network Address Translation

4.6.1 NAT66[11]

While NAT is not encouraged with IPv6, there may be some situations where it might be useful. Some customers may use NAT to translate between IPv4 and IPv6 networks. Some customers might want to translate IPv6 Unique Local addresses to IPv6 Global addresses using NAT66 in an attempt to preserve the present IPv4-accepted approach of using private addresses in the LAN. While there is a RFC 6296--**IPv6-to-IPv6 Network Prefix Translation**, it is a relatively new proposal and has not been widely adopted by network vendors.

One of the stated main benefits of IPv6 is theelimination ofthe need for NAT. However, during a transitional period, NAT66 may have a role to play in bridging between IPv4 and IPv6 networks.

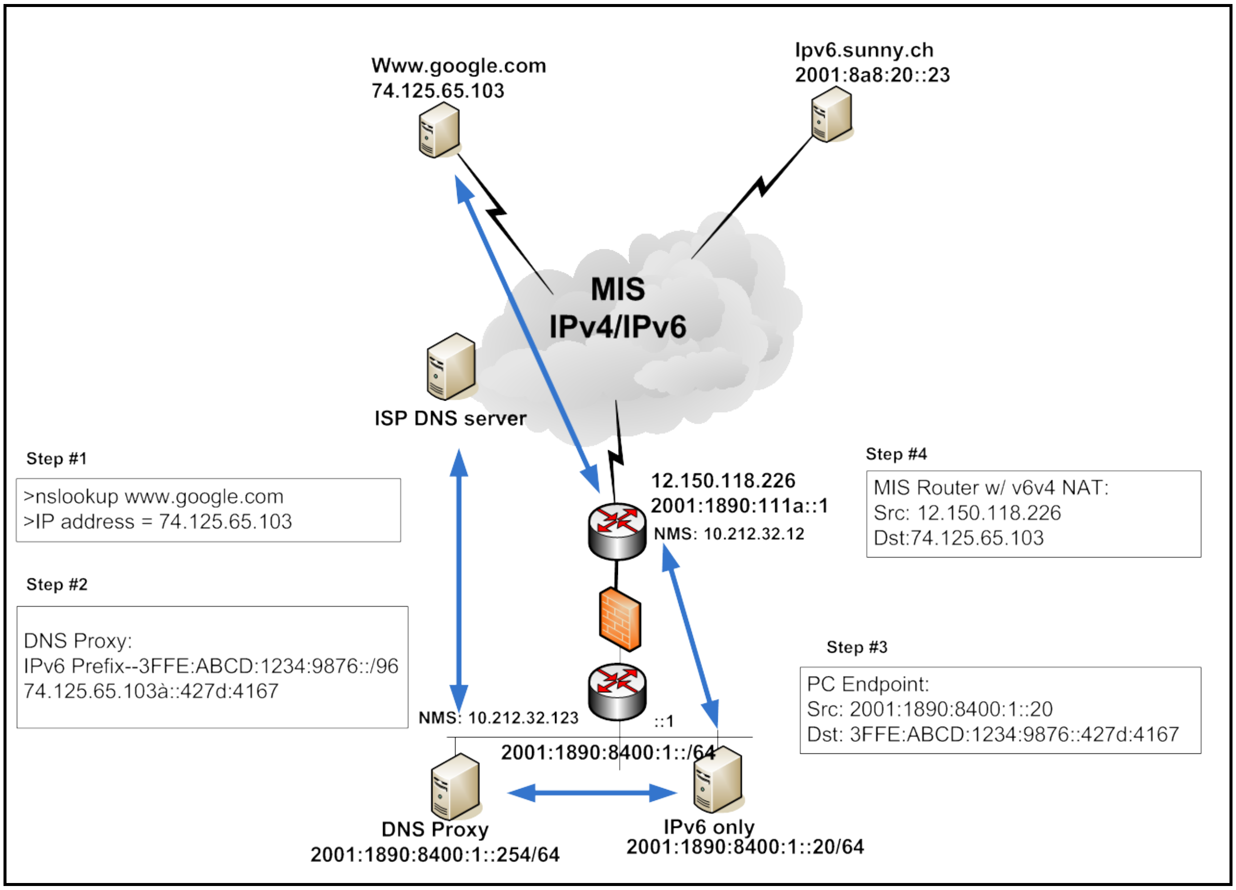
[11] NAT66 is a term used to describe network address translation of an IPv6 address to another IPv6 address.

So why is NAT66 even being considered? IPv6 is in its early stage of network adoption, and it's not clear how some customers will be able to deploy a multi-homed solution using IPv6 addresses allocated by their Service Providers (SP). Large companies will likely obtain their own Provider Independent (PI) IPv6 addresses directly from regional registries like ARIN and RIPE, but there is also a push to encourage customers to obtain addresses directly from their SPs.

With PI addresses, customers can advertise their address block over any provider's network. However SP-allocated addresses cannot be advertised across another SP's network. This means customers with multi-homed networks will not be able to failover one SP's address to another SP's network. In today's IPv4 networks, it is a widely accepted approach in designing a multi-homing solution to advertise a provider's address across another SP's network. This however is forbidden in IPv6 networks. It's not a technical limitation but rather a policy decision intended to control the number of routes that are advertised on the Internet. Without such a policy, IPv6 Internet routing tables could easily grow out of control and overwhelm Internet resources.

This policy may also have the opposite effect in that many companies who deploy multi-homing networks will likely opt for PI addresses thereby increasing the number of BGP routes on the IPv6 Internet. In order to avoid such scenarios, a viable multi-homing solution must exist. NAT66 in theory will allow outbound traffic to be translated to an available network's IPv6 address. So if a network fails, outbound traffic will be redirected to the surviving network and translated to the appropriate IPv6 address. Return traffic will follow the proper path back to the surviving network. Unfortunately, NAT66 does not resolve the issue for inbound traffic. If a network fails, then Internet traffic bound for the failed circuit must somehow learn about an alternate destination. NAT66 does not address this issue. Customer would need to incorporate the use of dynamic DNS advertisements to redirect traffic to the surviving network.

4.6.2 NAT 64[12]



Step #1

```
>nslookup www.google.com
>IP address = 74.125.65.103
```

Step #2

DNS Proxy:
IPv6 Prefix--3FFE:ABCD:1234:9876::/96
74.125.65.103à::427d:4167

Step #3

PC Endpoint:
Src: 2001:1890:8400:1::20
Dst: 3FFE:ABCD:1234:9876::427d:4167

Step #4

MIS Router w/ v6v4 NAT:
Src: 12.150.118.226
Dst:74.125.65.103

Figure 4.6.2—NAT64

Another NAT option that may be widely deployed is NAT 64. It is used to allow an IPv6-only WAN/LAN to access IPv4 resources. There are two critical pieces in order for NAT64 to work correctly: DNS64 proxy and NAT64 gateway. The purpose of a DNS64 proxy server is to translate an IPv4 address to "pseudo"-IPv6 address. In the example illustrated in Figure 4.6, it depicts a network scenario where corporate LAN users are attempting to access IPv4 internet resources. In this example, an IPv6-only endpoint wants to access www.google.com that is only available by IPv4 address. First, the IPv6-only endpoint sends a Quad-A or AAAA record request to the DNS64 proxy. Upon receiving the Quad A request, the DNS proxy sends a Standard A query to its upstream DNS servers for www.google.com's IPv4 address(step #1). The DNS proxy converts the address to IPv6 by appending the 32-bit address (in hex) to a preconfigured /96 prefix on the proxy server (step #2). The IPv6 endpoint uses the converted address as the destination address forwarding to the default gateway (step #3).

On the firewall or Internet gateway, the IPv6 converted address must be translated back to the original IPv4 address of www.google.com. This is done by stripping off the prepended /96 bits and converting the 32-bit hex into a more recognizable decimal IPv4 address of www.google.com. The IPv6 source address is also NAT'd to the gateway's IPv4 address allowing it to access www.google.com via IPv4. However, customers should keep in mind the ramifications of using NAT which utilizes valuable router resources in memory and CPU to maintain the mapping of IPv6 to IPv4 addresses. For situations where NAT uses an overload technique of multiplexing by port numbers, users must be aware that there is a limitation of

[12] NAT64 is term used to describe network address translation of an IPv6 address to an IPv4 address.

65,536 sessions that routers can support. With most web sites that contain links to multiple servers to obtain web contents like images and advertisements, 65,536 sessions could be easily consumed in a short time depending on the number of users at a given location.

Some customers may choose to deploy the NAT64 solution in a different way. Instead of translating outbound traffic as discussed above, some customers may use NAT64 for inbound traffic. Why would anyone do this? This approach allows customers to assign a single IPv4 address to their servers while being able to service IPv6 users. IPv6 headers will be translated to IPv4 headers before reaching the server. Many customers are concerned that IPv6 is not well understood or supported by their hardware/software vendors. If these servers are upgraded to dual-stack, it may introduce problems to the network, systems, and/or software. Therefore many customers are considering NAT64 solution (or Global Load Balancer) to translate IPv6 to IPv4 address for their application servers and avoid introducing uncertainty to a rather stable IPv4 system.

4.7 Proxy Server

Proxy servers are used by companies to control access to the IPv4 Internet. These organizations will likely continue to use proxy servers to access IPv6 websites. It is not automatic that existing IPv4 proxy servers will immediately support IPv6 proxy translation traffic. Customers should refer to their respective hardware vendor for more detail. This section assumes the proxy server is capable of handling web proxy requests for both IPv4 and IPv6 websites. As such, the outside interface (or interface facing the Internet) should be configured to support both IPv4 and IPv6 addresses while the LAN interface facing the corporate user is IPv4 only.

With this approach, internal corporate users will continue to point to the proxy server via an IPv4 address. If users need to access an IPv6 website, they can simply enter the URL in the browser, upon which the HTTP request is made to the proxy server with the URL of the destination IPv6 website. The proxy server then queries the DNS server for the IP address of the URL, and the content is retrieved and served back to the original endpoint who requested the site as an IPv4 address. The internal user is not aware that it was an IPv6 website. Therefore this solution has minimal impact to the corporate network. The only impacted resource is the proxy server as it is being serviced to support dual-stack on the interface facing the Internet. No changes are required to be pushed out to corporate users who can continue to point to the same proxy server via its IPv4 address. IPv6 to IPv4 translation is performed on the proxy server. However it’s important to note that this solution is appropriate for applications that are supported by a proxy server such as HTTP. If customers require IPv6 access for non-supported applications, then another IPv6 transition solution must be explored.

5 Summary

Adoption of IPv6 appears to be inevitable due to the exhaustion of IPv4 endpoint addresses. Besides additional address capacity there are numerous other significant changes in IPv6 that are discussed in this document. A key point to be understood is that IPv4 and IPv6 function as completely different networks. They can provide connectivity on the same physical interfaces but they cannot simply hand off packets between each other's protocol handling programs.

This document discusses IPv6 networking, addressing, and transition methods from IPv4 to IPv6. For IPv6 issues and configurations specific to AT&T networking services, the reader is referred to "AT&T VPN Dual-Stack (IPv4 / IPv6) Configuration Guide."