The Case for IPv6

The IPv6 internetwork protocol standard is of central importance to the continued viability of enterprise networks and the global Internet.

Introduction

The Internet Protocol (IP) has its roots in early military networks of the 1970s, but it's been within the past decade that IP has made its unstoppable conquest of the world's networks. Today, IP has established itself as the primary vehicle for our global system of electronic commerce, enabling a vast array of client/server and peer-to-peer computing applications. Although the IP success story took years to unfold, many in the networking community still express amazement and surprise that it happened at all, considering the strength of the opposing forces: OSI, SNA, DECnet, NetWare, et al. But what's even more amazing is the industry's dawning realization that it's time to make retirement plans for the current IP version (IPv4), a protocol that is a wizened crone compared to the advanced computers and network applications it supports.

In anticipation of the impending demise of IP as we know it, the Internet Engineering Task Force (IETF) has produced a comprehensive set of specifications (RFC 1752, 1883, 1886, 1971, 1993, etc.) that define the next-generation IP protocol known as "IPng," or "IPv6." IPv6 is both a near-term and long-range concern for network owners and service providers. On one hand, IPv6 products have already come to market; on the other hand, IPv6 work will likely continue well into the next decade. Though it is based on much-needed enhancements to IPv4 standards, IPv6 should be viewed as a broad retooling project that will ultimately provide a much-needed evolutionary rearchitecting of today's overstressed internetworks.

Because it lays the groundwork for the next era of networking, IPv6 is also of vital importance to businesses and network access providers of all sizes. It may sound like "just another protocol," but IPv6 is addressing performance, scalability, security, ease-of-configuration, and network management issues that are central to the ongoing competitiveness and bottom-line performance of all types of network-dependent businesses.

Organization of this Paper

The phased implementation of the next generation of IP is a topic of considerable significance to all those whose lives are tied in some way to IP networking. End users, industry executives, network administrators, protocol engineers, and many others will find that a working understanding of IPv6 provides a window to the future of internetworking and advanced distributed computing applications. The current document presents IPv6 issues in two parts:

- Part I: The Business Case for IPv6
- Part II: The Technical Case for IPv6

The first part gives a high-level view of business issues, protocol basics, and industry realities. The second part delves deeper into the functional and technical aspects of IPv6, including the aggregation-based allocation of the 128-bit address space, stateless autoconfiguration of IPv6 host populations, routing mechanics, new security features, and some real-world IPv4/IPv6 transition strategies. The message this report aims to convey is: If you are currently engaged in internetworking in any way, IPv6 is a critical-path network technology for you. The upgrade of IPv4 is a big project that covers a lot of territory, but it will positively impact your network, your bottom line, and your career. Learn about it sooner rather than later, for IPv6 product development and implementation efforts are already underway all over the world.

Part I: The Business Case for IPv6

Given the ever-increasing business requirements for interactive multimedia, and high-bandwidth network applications, IPv6 is critical to the continued viability of enterprise internetworks and the public Internet at large. Despite its importance and the efforts of some of the brightest minds in the network industry, the birth of IPv6 has been attended by a number of somewhat misleading myths and portrayals that can easily distract network owners who are in the process of crafting a forward-looking network strategy. Confusion is to be expected, considering the mammoth implications of migrating our global internetwork infrastructure to an updated protocol. But if the IPv6 myths are perpetuated indefinitely, there's a chance that the Internet will languish with a 20-year-old set of protocol components, while end-user and business requirements for advanced network services expand exponentially. It's time to clear the air.

Myth #1: The only driving force behind IPv6 is address space depletion.

Many of the discussions about a new Internet protocol focus on the fact that we will sooner or later run out of Network Layer addresses, due to IPv4's outdated 32-bit address space. The Internet Network Information Center (InterNIC) is the authority that assigns blocks of IP addresses to large network service providers and network operators. Since 1991, InterNIC has been increasingly stingy about the way these addresses are handed out, though most predictions for IPv4 address exhaustion target a time frame that starts well into the next decade.

With the long-haul in mind, IPv6 has been outfitted with an enormous 128-bit address space that should guarantee globally unique addresses for every conceivable variety of network device for the foreseeable future (i.e., decades). IPv6 has 16 bytes of addressing, or …

340,282,366,920,938,463,463,374,607,431, 768,211,456

… addresses, according to one IPv6 maven. The addressing gets a lot of attention but it is only one of many important issues that IPv6 designers have tackled. Other IPv6 capabilities have been developed in direct response to currently critical business requirements for more scalable network architectures, improved security and data integrity, integrated quality-of-service (QoS), autoconfiguration, mobile computing, data multicasting, and more efficient network route aggregation at the global backbone level. IPv6 is a big package, and addressing is only the most visible component of the work.

Myth #2: Extensions to IPv4 can replicate IPv6 functionality.

The many benefits of IPv6 will not come without a transition effort, which has prompted some in the industry to promote the idea of extending the life of IPv4 indefinitely with changes to the protocol standards and various proprietary techniques. One example of IPv4 extension is found in network address translators (NAT) that preserve IPv4 address space by intercepting traffic and converting private intra-enterprise addresses into globally unique Internet addresses. Other examples include the various quality-of-service and security enhancements to IPv4.

Although these extensions may prove valuable in certain limited and short-term scenarios, they ultimately will limit connectivity, interoperability, and performance in enterprises that are substantially network-dependent. In general, IPv4 extensions are no substitute for a protocol suite that has been designed from the ground up with scalable addressing, advanced routing,

By conservative estimates, IPv6 will support thousands of addresses for each square meter of the Earth's surface.

security, quality-of-service, and related features. Network owners need to be very wary of vendor claims that all the shortcomings of IPv4 can be addressed by extension products such as NAT and proprietary security gateways. The economics of IPv4 extensions can be gauged only when the resulting reduction in connectivity is taken into account. There is ultimately no substitute for IPv6 in organizations that need the high levels of internal and external connectivity that will be required by emerging multimedia, interactive, and transaction-oriented network applications. The future of such important business tools as intranets and the World Wide Web is closely tied to the availability of robust, advanced internetwork protocols.

Myth #3: IPv6 support for a large diversity of network devices is not an end-user or business concern.

Over the next few years, conventional computers on the Internet will be joined by a myriad of new devices, including palmtop personal data assistants (PDA), hybrid mobile phone technology with data processing capabilities, smart set-top boxes with integrated Web browsers, and embedded network components in equipment ranging from office copy machines to kitchen appliances. As new devices make their way onto the Internet, they will strain the existing network fabric in ways the early IP protocol designers could hardly have imagined. Some of the new devices requiring IP addresses and connectivity will be consumer-oriented, but a large number will also become integral to the information management functions of corporations and institutions of all sizes.

IPv6's 128-bit address space will allow businesses to deploy a huge array of new desktop, mobile, and embedded network devices in a cost-effective, managed manner. Further, IPv6's advanced autoconfiguration features will make it feasible for large numbers of devices to attach dynamically to the network, without incurring unsupportable costs for the administration for an ever-increasing number of adds, moves, and changes. The business requirement for IPv6 will be driven by end-user applications. To remain competitive in the coming era of high-density networking, businesses should exploit IPv6 to create a highly scalable address space and robust autoconfiguration services that will remain viable in the face of an explosion of end-user networking needs.

Myth #4: IPv6 is primarily relevant to backbone routers, not end-user applications.

It is true that IPv6 paves the way for efficient multitiered routing hierarchies that limit the uncontrolled growth of backbone router tables. But many of the advanced features of IPv6 also bring direct benefits to end-user applications at the workgroup and departmental levels. New IPv6 security features, for instance, give applications encryption and authentication services that are an integral part of the IP stack. For mobile business users and dynamic departmental staffs, the automatic configuration components of IPv6 will allow the efficient assignment of IP addresses without the delays and cost associated with manual address administration, which takes place in many current IP networks. IPv6's built-in quality-of-service features lay the groundwork for more deterministic end-to-end service levels in time-sensitive interactive and multimedia applications. IPv6 is very much both an end-user concern and a business concern.

Myth #5: Asynchronous Transfer Mode (ATM) cell switching will negate the need for IPv6.

ATM and other switching methods are extremely valuable technologies for present and future internetworks, but ATM is, by itself, not a replacement for today's packet routing, Internet architecture. Fortunately, network owners do not have to make a choice between ATM or IPv6 because the two protocols will continue to serve very

different and complementary roles in corporate networking. Large networks will undoubtedly make use of both protocols. For forward-thinking network designers, ATM is an ideal transmission medium for high-speed IPv6 backbone networks. And indeed, a great deal of standards and development work is being devoted to integrating ATM and IPv6 environments. IPv6, like its predecessor, provides Network Layer services over all major link types, including ATM, Ethernet, Token Ring, ISDN, Frame Relay, and T1.

Myth #6: IPv6 is something that only large telco companies or the government should worry about.

Some in the industry have characterized IPv6 as a concern that's outside the corporate network and outside the current time frame. In reality, IPv6 is an inside technology that is critical to the operations and continued efficiency of day-to-day business activities. But the only way that IPv6 will take hold and succeed is if businesses and institutions of all types come to terms with the inadequacies of IPv4 and begin to lay plans for migration. In the past few years, Internet protocols have enabled a whole new style of distributed commerce that brings people together inside enterprises and gives enterprises access to the entire world. Now it is up to the networking community to ensure that this success continues. And from a sheer business performance perspective, networked enterprises that invest in IPv6 planning now will have a decided competitive advantage as the information age proceeds.

IPv6: A Protocol Overview

IPv6, the Next-Generation Internet Protocol, was approved by the Internet Engineering Steering Group on November 17, 1994 as a Proposed Standard. Since that time a large number of end-user organizations, standards groups, and network vendors have been working together on the specification and testing of early IPv6 implementations. A number of IETF workgroups have defined IPv6 projects that are well underway, including the basic IPv6 protocol specification, address architectures, Domain Name Servers (DNS), security, transition mechanisms, and Internet Control Message Protocol (ICMP).

Standards work on IPv6 and related components is far enough along that vendors have already committed to a considerable number of development and testing projects. All of the major router vendors have committed to adding IPv6 to their products.

Endstation vendors such as Digital Equipment Corporation, Apple, Hewlett-Packard, Novell, and Sun Microsystems have likewise begun the task of delivering IPv6 on desktop machines and servers, as have major mainframe manufacturers. Many organizations are working on IPv6 drivers for the popular UNIX BSD operating environment. Network software vendors have announced a wide range of support for IPv6 in network applications and communication software products. A test bed called the 6Bone has been established, which currently links a large number of IPv6 end- and intermediate-node devices in North America, Europe, and the Pacific Rim.

IPv6 Design Goals

IPv6 has been designed to enable high-performance, scalable internetworks to remain viable well into the next century. A large part of this design process involved correcting the inadequacies of IPv4. It is only by delving into the full range of IPv6 improvements that the full benefits to enterprise and provider networks can be evaluated. Some of the qualities of IPv6 are found in obviously enhanced features, such as the larger address space and streamlined packet design. Other qualities are less tangible and relate to the fresh start that IPv6 gives to those who build and administer networks. With the clean slate that IPv6

provides, it will be possible to create a new, well-structured, efficient routing hierarchy to replace today's chaotic patchwork of addressing anomalies and legacy routes. The following sections give an overview of the obvious and not-so-obvious improvements that IPv6 brings to enterprise networking and the global Internet.

Addressing and Routing

IPv6 provides a framework for solving some critical problems that currently exist inside and between enterprises. On the global scale, IPv6 will allow Internet backbone designers to create a highly flexible and open-ended global routing hierarchy. At the level of the Internet backbone where major enterprises and Internet Service Provider (ISP) networks come together, it is necessary to maintain a hierarchical addressing system, much like that of the national and international telephone systems. Large central-office phone switches, for instance, only need a three-digit national area code prefix to route a long-distance telephone call to the correct local exchange. Likewise, the current IPv4 system uses a (somewhat haphazard) form of address hierarchy to move traffic between networks attached to the Internet backbone.

Without an address hierarchy, backbone routers would be forced to store routing table information on the reachability of every network in the world. Given the current number of IP subnets in the world and the growth of the Internet, this is not feasible. With a hierarchy, backbone routers can use IP address prefixes to determine how traffic should be routed through the backbone. IPv4 uses a technique called Classless InterDomain Routing (CIDR), which allows flexible use of variable-length network prefixes. With this flexible use of prefixes, CIDR permits considerable "route aggregation" at various levels of the Internet hierarchy, which means backbone routers can store a single routing table entry that provides reachability to many lower-level networks.

But the availability of CIDR routing does not guarantee an efficient and scalable hierarchy. In many cases, legacy IPv4 address assignments that originated before CIDR do not facilitate summarization. In fact, much of the IPv4 address space was formed before the current access provider hierarchy was developed. The lack of uniformity of the current hierarchical system, coupled with the rationing of IPv4 addresses, means that Internet addressing and routing increasingly are fraught with complications at all levels. These issues affect high-level service providers and individual end users in all types of businesses.

As Above, So Below

Many of the same problems that exist today in the Internet backbone are also being felt at the level of the enterprise and the individual business user. When an enterprise can't summarize its addresses, backbone routing tables can expand in a way that is ultimately unsupportable. If an enterprise can't present unique addresses to the Internet, it may be forced to deploy private, isolated address space that isn't visible to the Internet.

Users in private address spaces with non-unique addresses typically are limited by gateways and network address translators in their connectivity to the outside world. NAT services are meant to allow an enterprise to have whatever internal address structure it

desires, without concern for integrating internal addresses with the global Internet. The NAT device sits on the border between the enterprise and the Internet, converting private internal addresses to a smaller pool of globally unique addresses that are passed to the backbone and vice versa (see Figure 1).

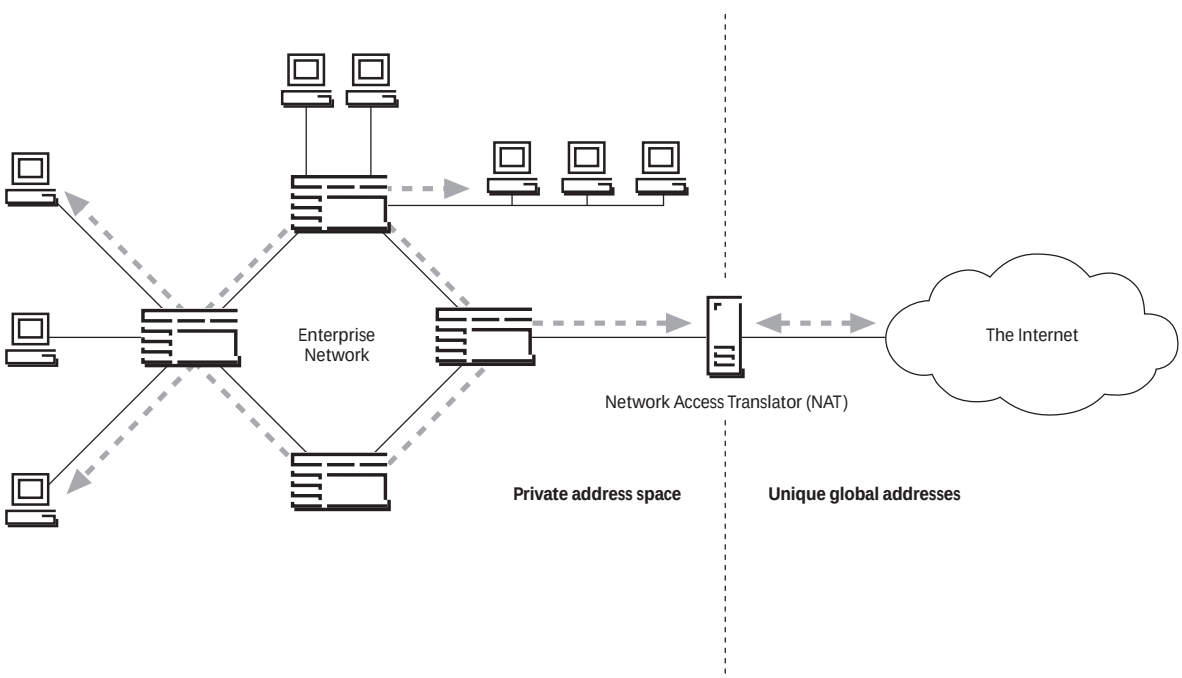
NAT may be appropriate in some organizations, particularly if full connectivity with the outside world is not desired. But for enterprises that require robust interaction with the Internet, NAT devices are not always desirable. The NAT technique of substituting address fields in each and every packet that leaves and enters the enterprise is very demanding, and can lead to a bottleneck between the enterprise and the Internet. A NAT may keep up with address conversion in a small network, but as Internet access increases, the NAT's performance must increase in a parallel fashion. The bottleneck effect is exacerbated by the difficulty of integrating and synchronizing multiple NAT devices within a single enterprise. It is highly unlikely that an enterprise will achieve the reliable high-performance Internet connectivity with NAT that is common today with multiple routers attached to an ISP backbone in an arbitrary mesh fashion.

NAT translators also run into trouble when applications embed their IP addresses in the packet payload, above the Network Layer. This is the case for a number of applications, including certain File Transfer Protocol (FTP) programs, and the Windows Internet Name Service (WINS) registration process of Windows 95 and Windows NT. Unless a NAT parses every packet all the way to the application level, it has no way of translating embedded addresses, which can lead to application failures. NAT can also wreak havoc with Domain Name Servers that work above the Network Layer. NAT services are a valuable tool for certain limited scenarios, but in general it may be hazardous to the long-term health of the Internet to promote NAT as a substitute for the comprehensive solution that IPv6 brings.

Return to Sender?

Another effect of IPv4's obsolescence relates to the ongoing need in many organizations to renumber stations. When an enterprise changes ISPs, it may have to either renumber all addresses to match the new ISP-assigned prefix, or implement address translation devices. Renumbering is also a reality for many corporations that undergo a merger or an acquisition that entails network consolidation.

Figure 1 | **Network Address Translator**



Address shortages and routing hierarchy problems increasingly are a threat to the network operations of larger enterprises, but they also affect small sites — even the isolated home worker who dials in to the office via the Internet. Smaller networks can be completely dropped from Internet backbone routing tables if they do not adhere to the address hierarchy. In the current system, ISPs with individual dial-in clients cannot allocate IP numbers as freely as they wish. Consequently, many dial-in users must use an address allocated from a pool on a temporary basis. In other cases, small dial-in sites are forced to share a single IP address among multiple end systems.

As peer-to-peer computing comes of age, a unique IP address lets end users gain direct connectivity to other users on the Internet to share a wide range of highly productive interactive applications, including real-time collaborative authoring, desktop-to-desktop video and audio, network white boards, and remote teaching. In general, today's environment of limited and poorly allocated addresses is already suboptimal, and it will degrade rapidly in the next few years as countless additional devices of all shapes and sizes are added to ISP rosters.

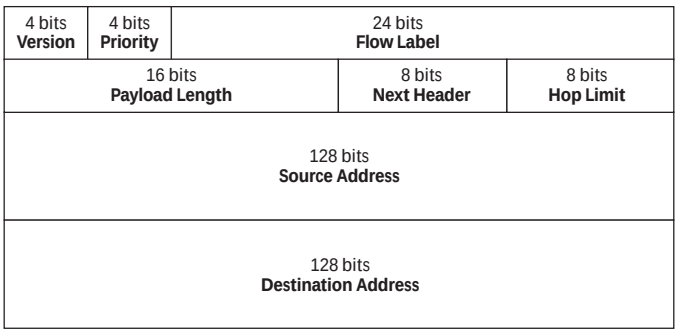
Enter IPv6

The large, flexible address space of IPv6 enables the definition of a flexible, hierarchical global routing architecture with many levels. An IPv6 address hierarchy can be aligned to geographic areas (like the U.S. telephone area code system), with allowances for the large backbone network topologies of provider networks that span geographic areas. (These will need network prefixes that aren't necessarily geographic.) Using CIDR-style flexible prefixes, the IPv6 address space can be allocated in a way that facilitates route summarization and controls expansion of route tables in backbone routers. IPv6 addressing means that large enterprises can avoid private address spaces indefinitely. It also means that ISPs will have enough addresses to allocate to smaller businesses and dial-in users that need globally unique addresses to fully exploit the Internet. In terms of the telephony metaphor: IPv6 addressing lets the network industry go beyond the current "party-line" era, which, for many of today's internetwork users, is similar to the early period of the phone industry, when residences had to share phone lines with neighbors.

Reducing Address Administration Workloads

IPv4 networks often employ the Dynamic Host Configuration Protocol (DHCP) to reduce the effort associated with manually assigning addresses to endstations. DHCP is termed a "stateful" address configuration tool because it maintains static tables that determine which addresses are assigned to new or moved stations. A new version of DHCP is being developed for IPv6 to provide similar stateful address assignment. IPv6 also adds a new dimension to autoconfiguration with a "stateless" address autoconfiguration service that does not require a manually configured server. Stateless autoconfiguration makes it possible for stations to configure their own addresses

Figure 2 | **IPv6 Packet Layout**



with the help of a local IPv6 router. Typically, the station combines its 48-bit MAC address with a network prefix it learns from a neighboring router.

The robust autoconfiguration capabilities of IPv6 will be a boon to internetwork users at many levels. When an enterprise is forced to renumber because of an ISP change, IPv6 autoconfiguration will allow hosts to be given new prefixes without manual reconfiguration of workstation or DHCP addressing. This function is also very useful on a smaller scale in enterprises that have trouble keeping up with the moves and changes of dynamic end-user populations. Autoconfiguration is even an important enabler of mobile computing because it allows mobile computers to receive valid IP addresses automatically, no matter where they connect to the network.

IPv6's Streamlined Format

IPv6 streamlines and enhances the basic header layout of the IP packet, improving greatly on IPv4 (see Figure 2). In IPv6, some of the IPv4 headers were dropped and others were made optional. The redesigned, simplified packet structure will, to some degree, offset the bandwidth cost of the longer IPv6 address fields. The 16-byte IPv6 addresses are four times longer than the 4-byte IPv4 addresses, but as a result of the retooling, the total IPv6 header size is only twice as large.

Beyond the streamlined packet format, IPv6 features improve support for header extensions and options, changing the way IP header options are encoded to allow more efficient forwarding. Optional IPv6 header information is conveyed in independent "extension headers" located after the IPv6 header and before the transport-layer header in each packet. Most IPv6 extension headers are not examined or processed by intermediate nodes (which was the case with IPv4). IPv6 header extensions are now variable in length and have less stringent length limits. IPv6 gives network software designers a very straightforward technique for introducing new header options in the future.

Option fields already have been defined for carrying explicit routing information created by the source node, as well as facilitating authentication, encryption, and fragmentation control. At the application level, header extensions are available for specialized end-to-end network applications that require their own header fields within the IP packet.

Native Security

Encryption, authentication, and data integrity safeguards are increasingly a standard aspect of enterprise internetworking. Traditionally, vendors in the IPv4 arena have been less than successful in adding robust security features to Network Layer components. This is largely due to the lack of interoperability caused by proprietary security extensions.

To correct this situation, IPv6 provides native data security capabilities that are based on its flexible header extensions. The authentication header extension to IPv6 ensures that a packet is actually coming from the host indicated in its source address. This authentication is particularly important to safeguard against intruders who configure a host to generate packets with forged source addresses. This type of source-address masquerading can spoof a server so that access may be gained to valuable data, passwords, or network control utilities. According to recent studies, IP spoofing is statistically one of the most common forms of deliberate intrusion, and with IPv4 there is no native way for a server to determine whether packets are being received from the legitimate endstation. Some enterprises have responded by putting proprietary firewalls in place, but these devices can introduce a number of new problems, including performance bottlenecks, restrictive network policies, and limited connectivity to the Internet.

The native authentication of IPv6 gives the industry a standards-based method to determine the authenticity of packets received at the Network Layer. Because the authentication headers in IPv6 are defined in IETF standards, it is highly likely that network products from different vendors will achieve interoperable authentication services. IPv6 implementations are required to support the MD5 algorithm for authentication and integrity checking, but since the specification is algorithm-independent, other techniques may be used as well. IPv6 authentication is particularly valuable where autoconfiguration is deployed. Without Network Layer authentication, network intruders may take advantage of DHCP and similar services to gain unassisted entry to a network. IPv6 authentication can ensure that illicit autoconfiguration does not take place.

Confidentiality and Privacy

Along with packet spoofing, another major hole in Internet security is the widespread deployment of traffic analyzers and network "sniffers," which can surreptitiously eavesdrop on network traffic. These generally helpful diagnostic devices can be misused by those seeking access to credit card and bank account numbers, passwords, trade secrets, and other valuable data. IPv4 provides no native data encryption scheme, so this must be accomplished in a less-than-interoperable manner, often at a higher layer.

IPv6 authentication headers do not provide privacy or confidentiality of data, so this is accomplished with another standard header extension that provides end-to-end encryption at the Network Layer. IPv6 encryption headers provide fields that carry encryption keys and other handshaking information, enabling interoperable encryption of the payloads in IP packets. IPv6 security headers can be used directly between hosts or in conjunction with a specialized security gateway that adds an additional level of security with its own packet signing and encryption methods.

Multicast and Anycast

One of the fastest growing business requirements for internetworks is the ability to transmit a stream of video, audio, news, financial, or other timely data to a group of functionally related but dispersed endstations. This is best achieved by Network Layer multicasting techniques. Typically, a server sends out a stream of multimedia or time-sensitive data that needs to be received by subscribers. A multicast-capable network can automatically replicate the server's packets and route them to each subscriber in the multicast group using an efficient path (see Figure 3). Routers use multicast protocols such as DVMRP (Distance Vector Multicast Routing Protocol) and MOSPF (Multicast Open Shortest Path First) to dynamically converge a packet distribution "tree" that connects all members of a group with the multicast server.

A new member becomes part of a multicast group by sending a "join" message to a nearby router. The distribution tree is then adjusted to include the new route. Multicast services mean that servers can send a single packet that will be replicated and forwarded through the internetwork to the multicast group on an as-needed basis. This conserves both server and network resources and, hence, is superior to unicast and broadcast solutions. Multicast applications are being developed for IPv4, but IPv6 extends IP multicasting capabilities by defining a very large multicast address space and a scope identifier that is used to limit the degree to which multicast routing information is propagated throughout an enterprise. Multicasting is an important feature of IPv6, and it actually replaces the IPv4 broadcast feature by supporting both functions.

Anyone for Anycast?

Anycast services are another innovation of the IPv6 specification that is not found in IPv4. Conceptually, anycast is a cross between unicast and multicast: Two or more interfaces on an arbitrary number of nodes are designated as an anycast group. A packet

Figure 3 | **Multicast in Action**

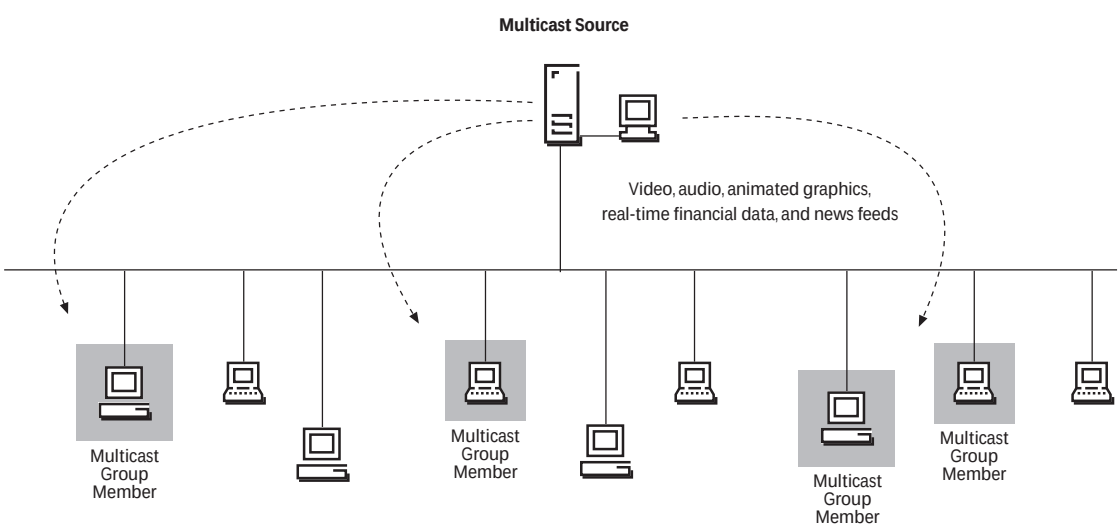
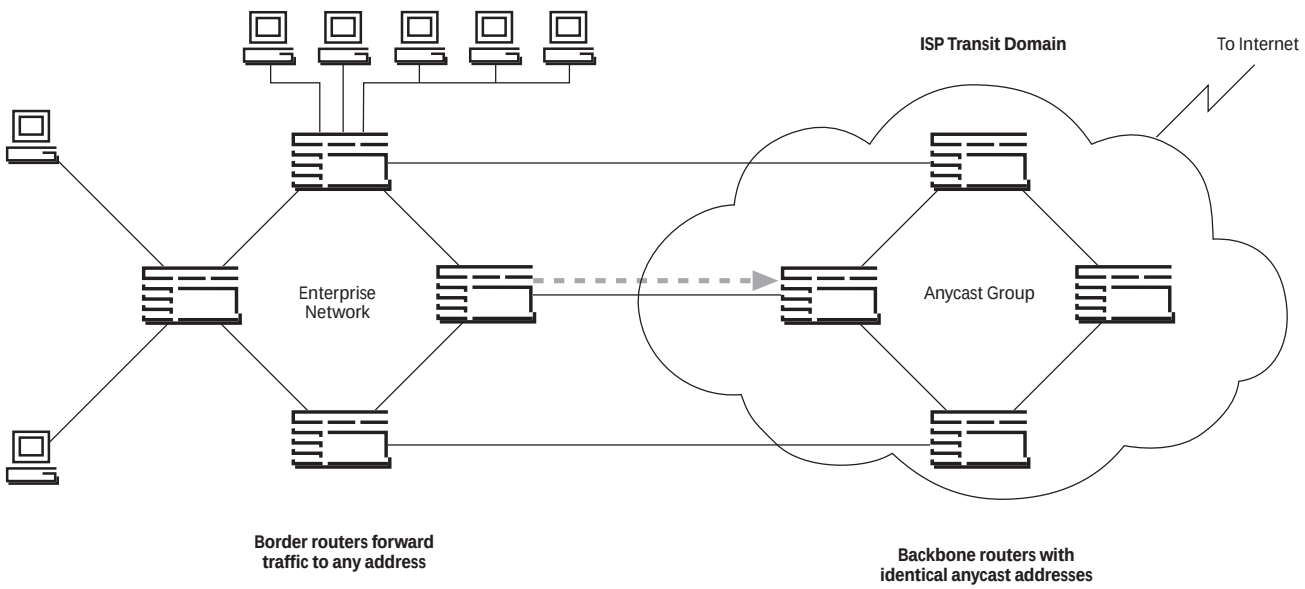


Figure 4 | **Anycast in Action**



addressed to the group's anycast address is delivered to only one of the interfaces in the group, typically the "nearest" interface in the group, according to current routing protocol metrics. This is in contrast with multicast services, which deliver packets to all members of the multicast group. Nodes in an anycast group are specially configured to recognize anycast addresses, which are drawn from the unicast address space.

Anycasting is a new service, and its applications have not been envisioned fully. Initially, it is recommended that anycast addresses be limited to intermediate nodes. This would allow, for example, an enterprise to use a single anycast address to forward packets to a number of different routers on its ISPs backbone (see Figure 4). If all of a provider's routers have the same anycast address, traffic from the enterprise will have several redundant access points to the Internet. And if one of the backbone routers goes down, the next nearest device automatically will receive the traffic. As anycast matures, it may become an important method for allowing endstations to efficiently access well-known services, mirrored databases, Web sites, and message servers. For instance, a corporation with several Lotus Notes servers could give interfaces on these devices the same anycast address. Packets from end-user Notes applications would be automatically forwarded to the nearest interface in the anycast group. Essentially, this is a highly flexible and cost-effective method of application load balancing. Anycast services could even be used to provide redundancy in the routing process by assigning all the DNS servers in an enterprise the same anycast address.

Quality of Service

The IPv6 packet format contains a new 24-bit traffic-flow identification field that will be of great value to vendors who implement quality-of-service network functions. Network-layer Quality of Service (QoS) products are still in the planning stage, but IPv6 lays the foundation so that a wide range of QoS functions may be made available in a highly open and interoperable manner.

In action, IPv6 flow labels can be used to identify to the network a stream of packets that needs special handling above and beyond the default, best-effort forwarding. Flow-based routing could give internetworks some of the deterministic characteristics associated with connection-oriented switching technology and telephony virtual circuits. For example, desktop video or audio streams could be given a flow label that tells routers they need a controlled amount of end-to-end latency. Flow labels can also be used to give traffic flows a specific level of security, propagation delay (e.g., satellite transmission), or cost. Experimental work with non-standard IPv4 QoS implementations has already shown that it is quite feasible to convey video and audio streams across the mesh internetwork topologies without excessive degradation. IPv6 paves the way for production application of this sort.

The Transition to IPv6

Few in the industry would argue with the principle that IPv6 represents a major leap forward for the Internet and the enterprises that rely on internetworking technology. IPv6 improves on IPv4 in many areas that are of great near-term and long-term value to network-dependent businesses. What is not agreed upon in the industry, however, is what shape and speed the transition from IPv4 to IPv6 will take. Some are lobbying for a wholesale, rapid adoption of IPv6 in the very near future. Others prefer to let the IPv6 project wait until address-space exhaustion and other issues force conversion. But given the magnitude of a migration that affects so many millions of network devices, it is clear that there will be an extended period when IPv4 and IPv6 will coexist at many levels of the Internet.

With the reality of extended IPv4/IPv6 coexistence looming, IETF protocol designers have expended a substantial amount of effort to ensure that hosts and routers can be upgraded to IPv6 in a graceful, incremental manner. Great pains have been taken to ensure that the transition will not entail large-scale obsolescence of IPv4 nodes or "fork-lift" upgrades for entire user populations in a short time frame. Transition mechanisms have been engineered to allow network administrators a large amount of flexibility in how and when they upgrade hosts and intermediate nodes. Consequently, IPv6 can be deployed in hosts first, in routers first, or, alternatively, in a limited number of adjacent or remote hosts and routers. The nodes that are upgraded initially do not have to be colocated in the same local area network or campus.

Another assumption made by IPv6 transition designers is the likelihood that many upgraded hosts and routers will need to retain downward compatibility with IPv4 devices for an extended time period (possibly years or even indefinitely). It was also assumed that upgraded devices should have the option of retaining their IPv4 addressing. To accomplish these goals, IPv6 transition relies on several special functions that have been built into the IPv6 standards work, including dual-stack hosts and routers and tunnelling IPv6 via IPv4.

The Dual-Stack Transition Method

Once a few nodes have been converted to IPv6, there is the strong possibility that these nodes will require continued interaction with existing IPv4 nodes. This is accomplished with the dual-stack IPv4/IPv6 approach. A great many hosts and routers in today's multivendor, multiplatform networking environment already support multiple network stack components. For instance, the majority of routers in enterprise networks are of the multiprotocol variety. Likewise, many workstations run some combination of IPv4, IPX, AppleTalk, NetBIOS, SNA, DECnet, or other protocols. The inclusion of one additional protocol (IPv6) on an endstation or router is a fairly trivial undertaking at the current time. When running a dual IPv4/IPv6 stack, a host has access to both IPv4 and IPv6 resources. Routers running both protocols can forward traffic for both IPv4 and IPv6 end nodes.

Dual-stack machines can use totally independent IPv4 and IPv6 addresses, or they can be configured with an IPv6 address that is IPv4-compatible. Dual-stack nodes can use conventional IPv4 autoconfiguration services (DHCP) to obtain their IPv4 addresses. IPv6 addresses can be manually configured in the 128-bit local host tables, or obtained via IPv6 stateless or stateful autoconfiguration mechanisms, when available. It is expected that major servers will run in dual-stack mode indefinitely, or until all active nodes are converted to IPv6.

IPv6 DNS

Domain Name Service is something that administrators must consider before deploying IPv6 or dual-stack hosts. The current 32-bit name servers cannot handle name-resolution requests for 128-bit addresses used by IPv6 devices. In response to this issue, IETF designers have defined an IPv6 DNS standard (RFC 1886, DNS Extensions to Support IP Version 6). This specification creates a new 128-bit DNS record type named "AAAA" (quad A) that will map domain names to an IPv6 address. Domain name lookups (reverse lookups) based on 128-bit addresses also are defined. Once an IPv6-capable DNS is in place, dual-stack hosts can interact interchangeably with IPv6 nodes. If a dual-stack host queries a DNS and receives back a 32-bit address, IPv4 is used; if a 128-bit address is received, then IPv6 is used. On sites where the DNS has not been upgraded to IPv6, hosts may resolve name-to-address mappings through the use of manually configured local name tables.

Applications that do not directly access the network stack will not need to be modified to run in the dual-stack environment. Network applications that directly interface with IP and related components will require updating if they are to use the IPv6 protocol. For example, applications that access the DNS must be enhanced with the capability to request the new 128-bit records — a fairly trivial change. Applications that exploit IPv6 security, quality of service, and other features will need more extensive updating.

Routing in IPv6/IPv4 Networks

Routers running both IPv6 and IPv4 can be administered in much the same fashion that IPv4-only networks are currently administered. IPv6 versions of popular routing protocols, such as Open Shortest Path First (OSPF) and Routing Information Protocol (RIP), are already under development. In many cases, administrators will choose to keep the IPv6 topology logically separate from the IPv4 network, even though both run on the same physical infrastructure. This

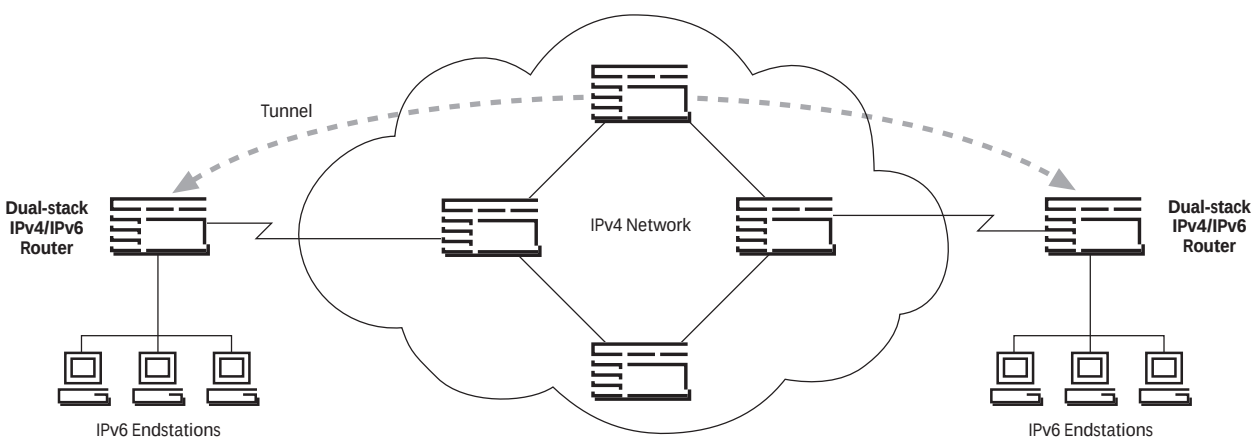
will allow the two to be administered separately. In other cases, it may be advantageous to align the two architectures by using the same domain boundaries, areas, and subnet organization. Both approaches have their advantages. A separate IPv6 architecture can be used to abolish the chaotic, inefficient IPv4 addressing systems with which many of today's enterprises suffer. An independent IPv6 architecture presents the opportunity to build a fresh, hierarchical network address plan that will greatly facilitate connection to one or more ISPs. This lays a foundation for efficient renumbering, route aggregation, and the other goals of an advanced internetwork routing hierarchy.

In most organizations where IPv6 is deployed incrementally, there is the strong possibility that all IPv6 hosts will not have direct connectivity to each other via IPv6 routers. In many cases there will be islands of IPv6 topology surrounded by an ocean of IPv4. Fortunately, IPv6 designers have fashioned transition mechanisms that allow IPv6 hosts to communicate over intervening IPv4 networks. The essential technique of these mechanisms is IPv6 over IPv4 tunnelling, which encapsulates IPv6 packets in IPv4 packets (see Figure 5).

Tunnelling allows early IPv6 implementations to take advantage of existing IPv4 infrastructure without any change to IPv4 components. A dual-stack router or host on the "edge" of the IPv6 topology simply appends an IPv4 header to each IPv6 packet and sends it as native IPv4 traffic through existing links. IPv4 routers forward this traffic without knowledge that IPv6 is involved. On the other side of the tunnel, another dual-stack router or host de-encapsulates the IPv6 packet and routes it to the ultimate destination using standard IPv6 protocols.

To accommodate different administrative needs, IPv6 transition mechanisms include two types of tunnelling: automatic and configured. To build configured tunnels, administrators manually define IPv6-to-IPv4

Figure 5 | **IPv6 over IPv4 Tunnelling**



address mappings at tunnel endpoints. On either side of the tunnel, traffic is forwarded with full 128-bit addresses. At the tunnel entry point, a router table entry is defined manually to dictate which IPv4 address is used to traverse the tunnel. This requires a certain amount of manual administration at the tunnel endpoints, but traffic is routed through the IPv4 topology dynamically, without the knowledge of IPv4 routers. The 128-bit addresses do not have to align with 32-bit addresses in any way.

Automatic Tunnelling

Automatic tunnels use "IPv4-compatible" addresses, which are hybrid IPv4/IPv6 addresses. Compatible addresses are created by adding leading zeros to the 32-bit IPv4 address to pad them out to 128 bits.

When traffic is forwarded with compatible addresses, the device at the tunnel entry point can automatically address encapsulated traffic by simply converting the IPv4-compatible 128-bit address to a 32-bit IPv4 address. On the other side of the tunnel, the IPv4 header is removed to reveal the original IPv6 address. Automatic tunnelling allows IPv6 hosts to dynamically exploit IPv4 networks, but it does require the use of IPv4-compatible addresses, which do not bring the benefits of the128-bit address space.

IPv6 nodes using IPv4-compatible addresses cannot take advantage of the extended address space, but they can exploit the other IPv6 enhancements, including flow labels, authentication, encryption, multicast, and anycast. Once a node is migrated to IPv6 with IPv4-compatible addressing, the door is open for a fairly painless move to the full IPv6 address space (hopefully with the help of an IPv6-based autoconfiguration service). IPv4-compatible addressing means that administrators can add IPv6 nodes while initially preserving their basic addressing and subnet architecture. Automatic tunnels are available when needed, but they may not be necessary in cases where major backbone routers are upgraded all at once to include the IPv6 stack. This is something that can be achieved quickly and efficiently when backbone routers support full remote configuration and upgrade capabilities (e.g., Bay Networks Backbone Node and Access Node routers).

It could be argued that IETF members are putting as much effort into transition as they are the basic IPv6 protocol specification. Whether or not this is true, the combination of tunnels, compatible addresses, and dual-stack nodes ensures that network administrators will have an enormous range of flexibility and interoperability when they deploy IPv6. Transition services allow network-dependent organizations to take advantage of the rich array of more technical IPv6 features, many of which are discussed in Part II of this document.

Part II: The Technical Case for IPv6

Tale of Two Headers

A good way to start an in-depth investigation of IPv6 is to compare the new streamlined IPv6 header with the current IPv4 header. Both headers carry version numbers and source/destination addresses, but as Figure 6 shows, the IPv6 header is considerably simplified, which makes for more efficient processing by routing nodes. Whereas the IPv4 headers are variable in length, all IPv6 headers have a fixed length of 40 bytes. This allows router software designers to optimize the parsing of IPv6 headers along fixed boundaries. Additional processing efficiencies have been realized by reducing the number of required header fields in IPv6. The classic IPv4 header contains 14 fields, whereas IPv6 only requires 8 fields.

One of the first IPv4 components to be discarded was the header length field, which is clearly no longer required due to the fixed header length of all IPv6 packets. The total length field of IPv4 has been retained in the guise of the IPv6 payload length field. But this field does not include the length of the IPv6 header, which is always assumed to be 40 bytes. The new payload length field can accommodate packets up to 64 KB in length. Even larger packets, called "jumbograms", will be forwarded by IPv6 routers if the payload length field is set to zero and a special extension header is added, as discussed below.

The time-to-live field of IPv4 has been given a face-lift in the form of the IPv6 hop limit field. Although the names are different, both fields are used by routers to decrement a maximum hop value by 1 for each hop of the end-to-end route. The hop-limit field is set to the appropriate value by the source node. When the value in the hop limit field is decremented to zero, the packet is discarded. The IPv6 hop-count field will store a value of up to 8 bits or 255 hops, which should be more than adequate for even the largest of networks for the foreseeable future.

In addition to the header length field, a number of basic IPv4 fields were eliminated from IPv6: type-of-service, fragment offset, identification, flags, checksum, and header length. The functionality of the IPv4 type-of-service field has been transferred to the two new IPv6 fields: flow control and priority. The

Figure 6 | **IPv4 and IPv6 Header Formats**

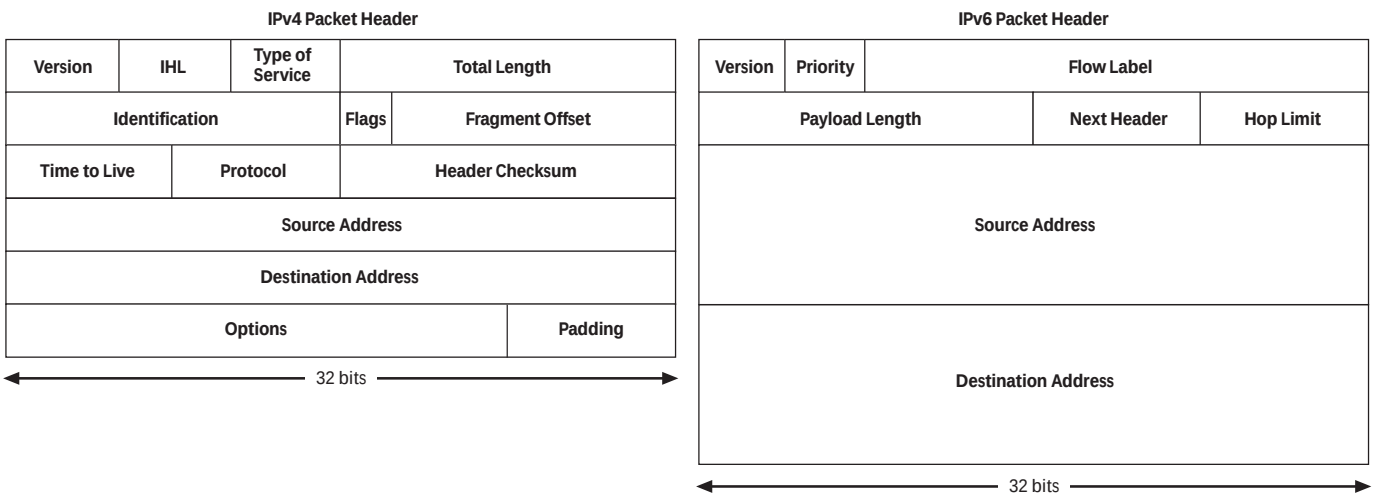


Figure 7 | **IPv6 Extention Headers**

IPv4 fragmentation fields (offset, identification, and flags) have been made optional headers in IPv6, as is discussed below. Finally, the IPv4 checksum fields have been abandoned in IPv6, due to the prevalence of error checking at other levels of the protocol stack. It is assumed that bad packets will be detected below, at the link layer, or above, at transport or higher layers.

Those Exceptional Extension Headers

To allow IPv4 packet headers the flexibility to carry optional information relevant to the routing process or host applications, IPv4 headers included an options field. This little-used field is carried by all IPv4 packets and is meant to convey information about security, source routing, and other optional parameters. The IPv4 options field has been replaced in IPv6 by flexible extension headers that travel after the primary IPv6 header and before the transport header and application payload. IPv6 extension headers are optional and provide a powerful means to support security, fragmentation, source routing, network management, and many other functions. An IPv6 packet can carry virtually any number of extension headers between the initial header and the higher layer payload. Figure 7 shows encryption and fragmentation headers travelling after the primary IPv6 header and before the Transmission Control Protocol (TCP) header.

The IPv6 extension header architecture replaces the IPv4 options field and also impacts the protocol type field, which is currently used to indicate the type of protocol within the datagram's payload, e.g., TCP or User Datagram Protocol (UDP). IPv6 replaces the protocol type field with a next header field that indicates the protocol carried in the next extension or payload header (e.g., a TCP/UDP header or a IPv6 optional header).

The IPv6 standards groups have already defined a number of extension headers and have also created a suggested (but not mandatory) guideline for the order of header insertion.

The suggested order for extension headers is as follows:

(Primary IPv6 header)

Hop-by-Hop options header
Destination options header-1
Source Routing header
Fragmentation header
Authentication header
IPv6 Encryption header
Destination options header-2

(Upper-layer headers)
(Payload)

Each extension header typically occurs only once within a given packet, except for the destination header, as explained on the following page.

Hop-by-Hop Options Header When present, this header carries options that are examined by intermediate nodes along the forwarding path. It must be the first extension header after the initial IPv6 header. Since this header is read by all routers along the path, it is useful for transmitting management information or debugging commands to routers. One currently defined application of the hop-by-hop extension header is the Router Alert option, which informs routers that the packet should be processed completely by a router before it is forwarded to the next hop. An example of such a packet is an RSVP's resource reservation message.

Destination Options Headers There are two variations of this header, each with a different position in the packet. The first incidence of this field is for carrying information to the first destination listed in the IPv6 address field. This header can also be read by a subsequent destination listed in the source routing header address fields. The second incidence of this header is used for optional information that is only to be read by the final destination. For efficiency, the first variation is typically located towards the front of the header chain, directly after the hop-by-hop header (if any). The second variation is relegated to a position at the end of the extension header chain, which is typically the last IPv6 optional header before transport and payload.

Source Routing Header The IPv6 routing extension header is an incarnation of the source routing function supported currently by IPv4. This optional header allows a source node to specify a list of IP addresses that dictate what path a packet will traverse. IETF RFC 1883 defines a version of this routing header called "Type 0," which gives a sending node a great deal of control over each packet's route. Type 0 routing headers contain a 24-bit field that indicates how intermediate nodes may forward a packet to the next address in the routing header. Each bit in the 24-bit field indicates whether the next corresponding destination address must be a neighbor of the preceding address (1 = strict, must be a neighbor; 0 = loose, need not be a neighbor).

When routing headers are used for "strict" forwarding, a packet visits only routers listed in the routing header (see Figure 8). In "loose" forwarding, unlisted routers can be visited by a packet. So if routers B and C are listed as strict but are not adjacent to each other (i.e., in order to get from B to C, a packet must pass some other router), packets will be dropped at B. This is a valuable feature when security and traffic

Figure 8 **Source Routing Extention Header**

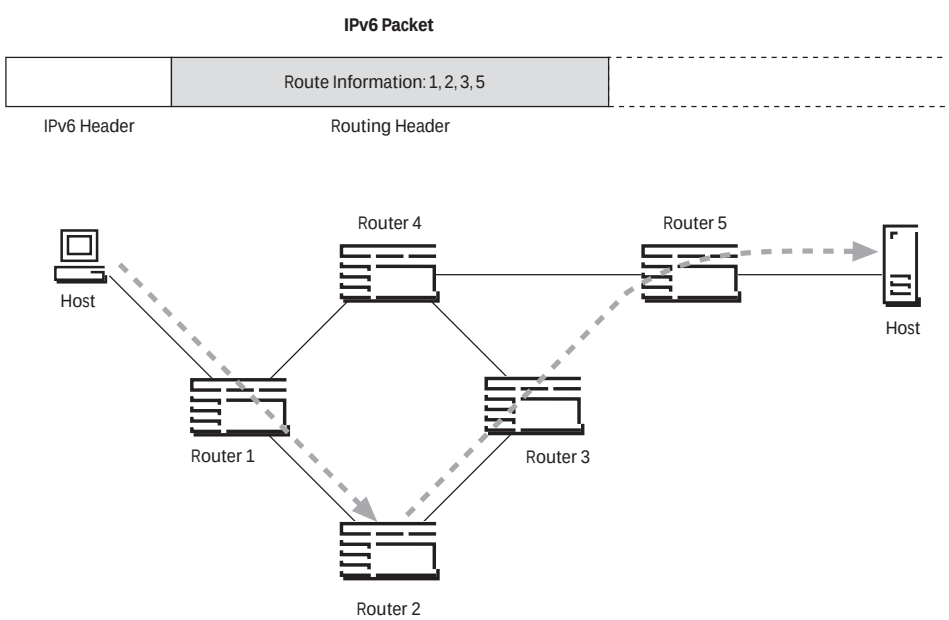
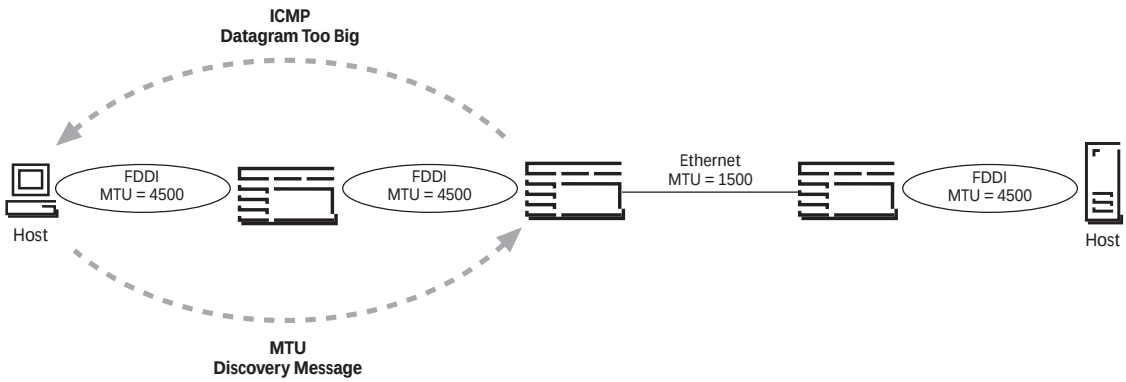


Figure 9 | **MTU Discovery Process**



control require that packets take a rigidly defined path. The strict/loose feature works in conjunction with another routing header field that contains a value equal to the total number of segments remaining in the source route. Each time a hop is made, this "segments left" field is decremented.

When Type 0 routing headers are used, the initial IPv6 header contains the destination addresses of the first router in the source route, not the final destination address. At each hop, the intermediate node replaces this destination address with the address of the next routing node, and the "segments left" field is decremented.

Fragmentation Header IPv4 has the ability to fragment packets at any point in the path, depending on the transmission capabilities of the links involved. This feature has been dropped in IPv6 in favor of end-to-end fragmentation/reassembly, which is executed only by IPv6 source and destination nodes. Packet fragmentation is not permitted in intermediate IPv6 nodes. The elimination of the fragmentation field allows a more streamlined packet and better router performance for the majority of cases where fragmentation is not required. Today's networks generally support frame sizes that are large enough to carry typical IP packets without fragmentation. In the event that fragmentation is required, IPv6 provides an optional extension header that is used by source nodes to divide packets into an arbitrary number of smaller units. The IPv6 fragmentation header contains fields that identify a group of fragments as a packet and assigns them sequence numbers. Because IPv6 routers do not fragment packets between end nodes, the responsibility for sending the correct size packet is with the source node, which needs to determine the Maximum Transmission Unit (MTU) of the links in the end-to-end path. For instance, if two FDDI networks with 4500-byte MTUs are connected by an Ethernet with an MTU of 1500, then the source station must send

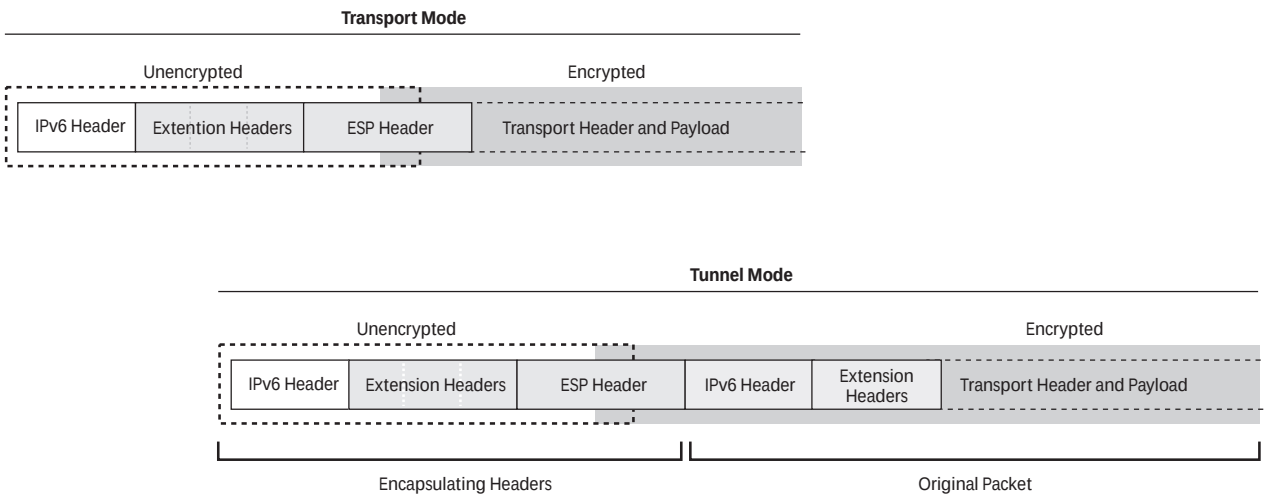
packets that are no larger than 1500. If higher level applications are using larger payloads, the source node can make use of the IPv6 fragmentation extension header to divide large packets into 1500-byte units for network transmission. The IPv6 destination node will reassemble these fragments in a manner that is transparent to upper layer protocols and applications.End nodes performing fragmentation can determine the smallest MTU of a path with the MTU path discovery process (e.g., RFC1191; see Figure 9). Typically, with this technique, the source node sends out a packet with an MTU as large as the local interface can handle. If this MTU is too large for some link along the path, an ICMP "Datagram too big" message will be sent back to the source. This message will contain a packet-too-big indicator and the MTU of the affected link. The source can then adjust the packet size downward (fragment) and retransmit another packet. This process is repeated until a packet gets all the way to the destination node. The discovered MTU is then used for fragmentation purposes. Although source-based fragmentation is fully supported in IPv6, it is recommended that network applications adjust packet size to accommodate the smallest MTU of the path. This will avoid the overhead associated with fragmentation/reassembly on source and destination nodes.

Authentication Header The current lack of a standardized network-layer security scheme is one of the most glaring deficiencies of the IPv4 Internet. Regular press reports of hackers spoofing servers and snooping data streams have become a constant reminder of the damage that can be done to IP-based corporate networks. The IPv6 standard addresses this situation with two important extension headers, one that enables the authentication of IP traffic for security purposes, and another that fully or partially encrypts IP packets. Implementation of security at the IP level can benefit "security aware" applications, as well as "security ignorant" applications that don't take explicit advantage of security features.

The IPv6 authentication extension header gives network applications a guarantee that the packet did in fact come from an authentic source. This combats the increasingly common occurrence of hackers configuring an IP host to impersonate another, to gain access to secure resources. Such spoofing can be used to obtain valuable financial and corporate data and can give persons outside the enterprise control of servers for malicious purposes. With IPv6 authentication headers, hosts establish a standards-based security association that is based on the exchange of algorithm-independent secret keys (e.g., MD5).

In a client/server session, for instance, both the client and the server need to have knowledge of the key. Before each packet is sent, IPv6 authentication creates a checksum based on the key combined with the entire contents of the packet. This checksum is then re-run on the receiving side and compared. This approach provides authentication of the sender and guarantees that data within the packet has not been modified by an intervening party. Authentication can take place between clients and servers or client and clients on the corporate backbone. It can also be deployed between remote stations and corporate dial-in servers to ensure that the perimeter of the corporate security is not breached.

Figure 10 | **Tunnel Mode and Transport Mode of IPv6 Encryption**

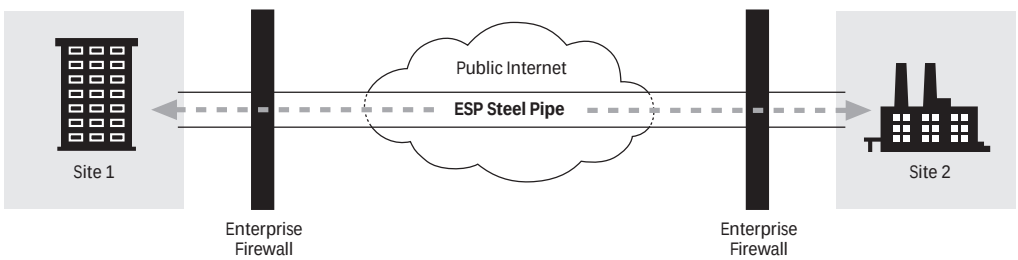


IPv6 Encryption Header Authentication headers eliminate a number of host spoofing and packet modification hacks, but they do not prevent the nondisruptive reading (sniffing, snooping) of the content of packets as they traverse the Internet and corporate backbone networks. This is the area addressed by the Encapsulating Security Payload (ESP) service of IPv6 — another optional extension header. Packets protected by the ESP encryption techniques can have very high levels of privacy and integrity — something that is not widely available with the current Internet, except with certain secure applications (e.g., private electronic mail and secure HTTP Web servers). ESP provides encryption at the network layer, making it available to all applications in a highly standardized fashion.

IPv6 ESP is used to encrypt the transport-layer header and payload (e.g., TCP, UDP), or the entire IP datagram. Both these methods are accomplished with an ESP extension header that carries encryption parameters and keys end-to-end. When just the transport payload is to be encrypted, the ESP header is inserted in the packet directly before the TCP or other transport header. In this case, the headers before the ESP header are not encrypted and the headers and payload after the ESP header are encrypted. This is referred to as "transport-mode" encryption. If it is desirable to encrypt the entire IP datagram, a new IPv6 and an ESP header are wrapped around all the fields (including the initial address fields) of the packet. Full datagram encryption is sometimes called "tunnel-mode" encryption because the contents of the datagram are only visible at the endpoints of the security tunnel (see Figure 10).

Fully encrypted datagrams are somewhat more secure than transport mode encryption because the headers of the fully encrypted packet are not available for traffic analysis.

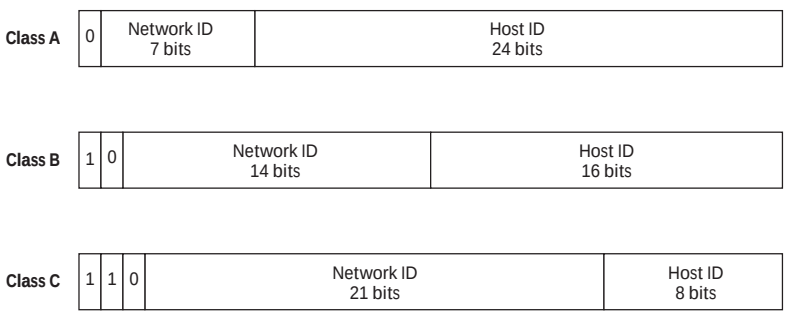
Figure 11 | **Firewalls and Steel Pipe**



For instance, full tunnel-mode encryption allows the addresses contained in IPv6 source routing headers to be hidden from packet sniffing devices for the public portion of a path. There is a considerable performance penalty for full encryption, due to the overhead and processing cost of adding an additional IPv6 header to each datagram. In spite of its cost, full ESP encryption is particularly valuable to create a security tunnel (steel pipe) between the firewalls of two remote sites (see Figure 11). The full datagram encryption in the tunnel ensures that the various headers and address fields of encrypted packets will not be visible as traffic traverses the public Internet. Within the tunnel, only the temporary encapsulating address header is visible. Once through the tunnel and safely within a firewall, the leading ESP headers are stripped off and the packet is again visible, including any source routing headers required to finish the path.

The encryption and authentication services of IPv6 work hand-in-hand to create a flexible and powerful security solution. In some cases an authentication header will be carried inside a fully encrypted or partially encrypted datagram, providing an additional layer of data integrity and verification of the sender's identification. In other cases, the authentication header may be placed in front of the encrypted transport-mode portion of the packet. This approach is desirable when the authentication takes place before decryption on the receiving end, which is the logical order in many cases. Taken together, the authentication and encryption services of IPv6 provide a robust, standards-based security mechanism that will play a critical role in the continuing expansion of commerce and corporate operations onto IP-based network fabrics.

Figure 12 **IPv4 Address Classes**



The IPv6 Address Architecture

Much of the discussion of IPv4 versus IPv6 focuses on the relative size of the address fields of the two protocols (32 bits versus 128 bits). But an equally important difference is the relative abilities of IPv6 and IPv4 to provide an advanced hierarchical address space that facilitates efficient routing architectures. IPv4 was initially designed with a class-based address scheme (see Figure 12), which divided address bits between network and host but did not create a hierarchy that would allow a single high-level address to represent many lower-level addresses. Hierarchical addressing systems work in much the same way as telephony country codes or area codes, which allow long-haul phone switches to efficiently route calls to the correct country or region using only a portion of the full phone number.

As the Internet grows, the non-hierarchical nature of the original IPv4 address space is proving to be increasingly inadequate.

The limitations of IPv4 addressing are currently hampering both the local and global levels of internetworking. To combat IPv4 deficiencies at the local area network level, the subnetting technique has been developed to create a more granular division of large networks. With subnet addressing, a single network address can stand for a number of physical networks, which conserves address space considerably (e.g., a single Class C address can be used to access several physical networks).

At the level of large internet backbones and global routing, IPv4 addresses can be more efficiently aggregated with supernetting, a form of hierarchical addressing. With supernetting, backbone routers store a single address that represents the path to a number of lower level networks. This can considerably reduce the size of routing tables in backbone routers, which increases backbone performance and lowers the amount of memory and number of processing routers required. Subnetting and supernetting have been particularly useful in extending the viability of the IPv4 Class C addresses. Both of these techniques are made possible by pairing addresses stored in routers with bit masks that indicate which bits in an address are valid at the various levels of the hierarchy.

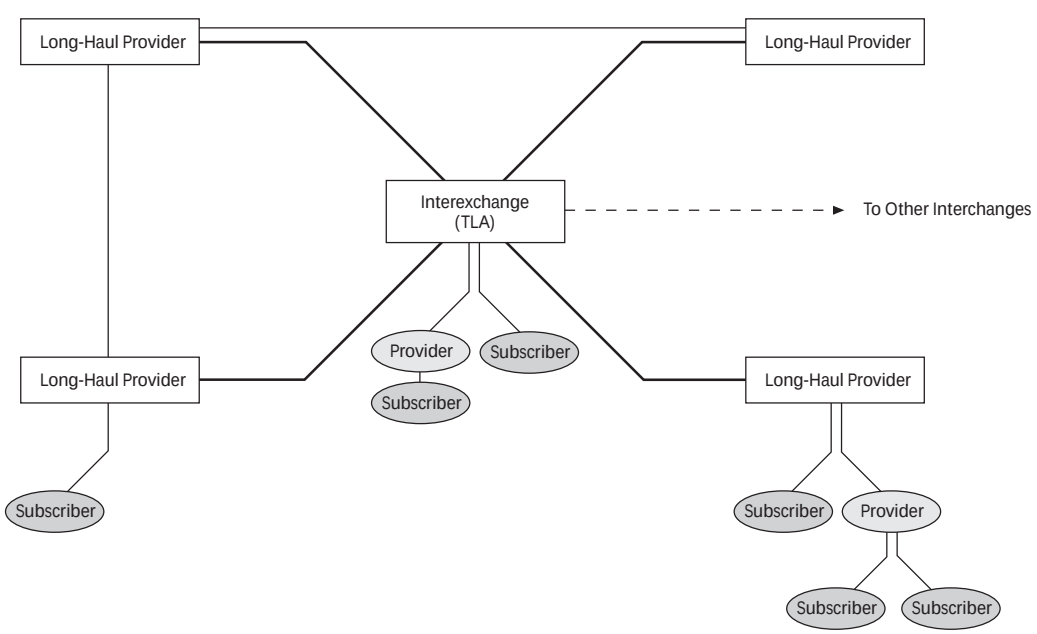
The process of creating an IPv4 routing hierarchy was formalized in Classless Interdomain Routing (CIDR) which uses bit masks to allocate a variable portion of the 32-bit IPv4 address to network, subnet, or host. For instance, CIDR allows a number of (plentiful) Class C addresses to be summarized by a single prefix address, allowing Class C addresses to function in a similar way to hard-to-get Class A and Class B addresses. CIDR has extended the life of IPv4 and helped the Internet scale to its current size, but it has not been implemented in a consistent way across the Internet and enterprise networks. Consequently, the routing table efficiencies and address space conservation advantages of CIDR are not today fully realized, nor will they ever be fully realized, due to the legacy nature of IPv4 networks and the difficulty of restructuring them. IPv4 will continue to waste its already inadequate address space as it continues to burden routers with inefficient routes and excessively large routing tables.

Yet another downside of IPv4 is found at the departmental and workgroup level of internetworking, in the high administrative workload associated with maintaining subnet bit masks and host addresses within the subnet structure, particularly where there are large, dynamic populations of end users. When an end user is moved in the subnetting environment, careful attention must be paid to ensure that the host renumbering process does not disrupt connectivity at any level of the stack. The complexities and pitfalls of current subnetting methods can eventually make IPv4 less than viable in large organizations that experience ongoing growth of internetwork user populations.

The IPv6 Address Hierarchy

In a direct response to the experience gained from IPv4, IPv6 has been designed from the ground up to provide a highly scalable address space that can be partitioned into a flexible and efficient global routing hierarchy. At the top of this hierarchy, several international registries assign blocks of addresses to top level aggregators (TLA). These TLAs are essentially the public transit points (exchanges) where long-haul providers and telcos establish peer connections — for example, MAE on the East Coast of the U.S.A., and Telehouse in London, England (see Figure 13). TLAs allocate blocks of addresses to Next Level Aggregators (NLA), which represent large

Figure 13 **Aggregation-based Allocation Structures**



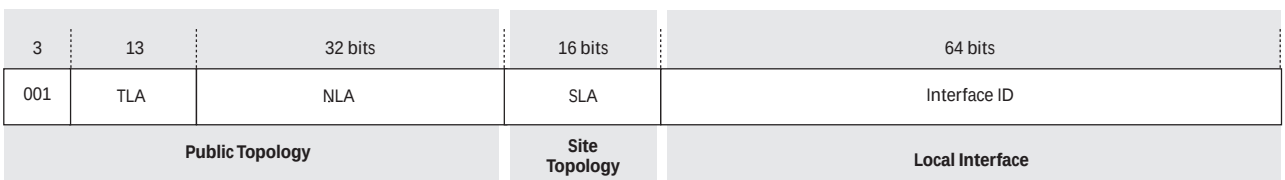
providers and global corporate networks. When an NLA is a provider, it further allocates its addresses to its subscribers. Routing is efficient because NLAs that are under the same TLA will have addresses with a common TLA prefix. Subscribers with the same provider have IP addresses with an NLA common prefix.

Although a number of allocation schemes are possible within IPv6's huge address space, an aggregation-based hierarchy is favored by IETF designers because it combines the advantages of provider and geographic allocation approaches. Provider allocation divides the hierarchy along lines of large service providers, regardless of their location. Geographic allocation divides the hierarchy strictly on the basis of the location of providers/subscribers (as does the telephony system of country and area codes). But both of these approaches have their drawbacks because large backbone networks often don't conform strictly to geographic or provider boundaries. Some large networks, for instance, may connect to several ISPs. And many large networks span numerous countries and geographical regions.

Aggregation-based allocation is based on the existence today of a limited number of high-level exchange points, where large long-haul service providers and telco networks interconnect. The use of these exchange points to divide the IPv6 address hierarchy has a geographical component because exchanges are distributed around the globe. It also has a provider orientation because all large providers are represented at one or more exchange points.

As shown in Figure 14, the first 3 address bits indicate what type of address follows (unicast, multicast, etc.). The next 13 bits are allocated to the various TLAs around the world. The following 32 bits are allocated to the next lower level of providers and subscribers.

Figure 14 **Aggregation-based IPv6 Addresses**

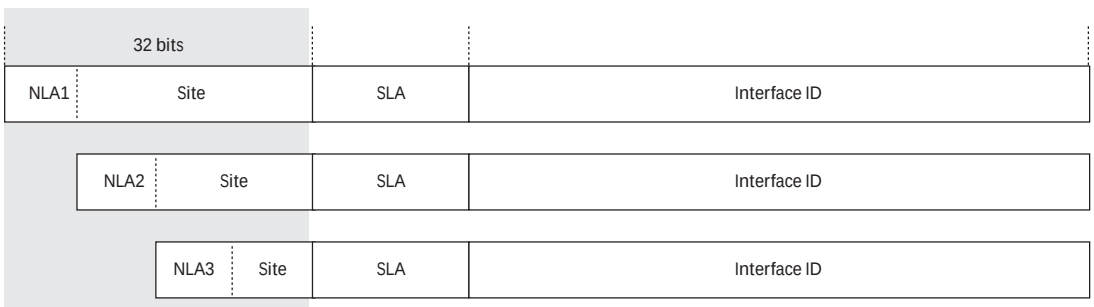


Next level aggregators can divide the NLA address field so as to create their own hierarchy, one that maps well to the current ISP industry, in which smaller ISPs subscribe to higher level ISPs, and so on. This is accomplished by the ongoing subdivision of the 32-bit NLA field (see Figure 15).

Following the NLA ID are fields for subscriber site networking information: Site Level Aggregator (SLA) and Interface ID. Typically, service providers supply subscribers with blocks of contiguous addresses, which are then used by individual organizations to create their own local addressing hierarchy and identify subnets and hosts. The 16-bit SLA field supports up to 65,535 individual subnets. The 64-bit Interface ID, which is used to identify an IPv6 interface on a network link, will typically be derived from the installed IEEE LAN adapter address.

Today's Internet backbone routers must maintain up to 40,000 or more routes. As the Internet continues to scale, IPv6's uniform application of hierarchical routing will likely be the only viable method for keeping the size of backbone router tables under control. With an aggregator-based address hierarchy, all of a subscriber's internal network segments can be reached through one or more high-level aggregation points. This allows backbone routers around the globe to efficiently summarize the routes to a customer's networks with high-level TLA address prefixes. Forwarding routes in highest level backbones can be quickly calculated by looking only at the TLA portion of the address. IPv6's large hierarchical address space also allows a more decentralized approach to IP address allocation. Service providers can allocate addresses independently from central authorities, encouraging global network growth and eliminating bureaucratic bottlenecks in the growth process.

Figure 15 | **Subdividing the NLA Address Space**



Aggregation-based addresses are just part of the total address space that has been defined for IPv6. Other address ranges have been assigned to multicasting and to nodes that only require unique addressing within a limited area (site-local and link-local addresses).

Site- and link-local addresses are available for private, internal use by all enterprises, and are not allocated by public registry authorities. Site-local addresses are a flexible way for networks to start off with non-unique local addresses that are later made globally unique by adding a prefix. This has an advantage: if an ISP changes, site local addressing can remain the same because it is not directly interfaced to the outside world. Link local addresses can be used for applications that are limited in scope to a single link, and also for temporary "bootstrapping" of stations before they receive a globally unique address (more on this in the section below).

Host Address Configuration

IPv6 clearly has a large enough address architecture to accommodate Internet expansion for decades to come. But the usefulness of IPv6 addresses will be severely limited if they are not matched with equally advanced configuration and management services. Fortunately, there is a great deal of work underway to ensure that IPv6 hosts can have their addresses automatically configured and reconfigured in a cost-effective and manageable way. Automatic address configuration is a very necessary component of hierarchical routing fabrics because it supports cost-effective numbering and renumbering of large populations of IP hosts.

Autoconfiguration capabilities are important whether provider-based or geographic address allocation is in effect. Occasionally, it may be necessary to renumber every host within an organization, as would be the case with a company that relocated its operations (with geographic addressing) or changed to another service provider (with provider-based addressing). Configuration of IP addresses is a constant fact of life at the workgroup and department levels of large networked organizations. IP addresses need to be configured for new hosts, for hosts that change location, and for hosts connected to physical networks that receive address modification (e.g., a new prefix). In addition to these traditional requirements for configuration, new requirements are emerging as large numbers of hosts become highly mobile.

The process of autoconfiguration under IPv6 starts with the Neighbor Discovery (ND) protocol. ND combines and refines the services provided in the IPv4 environment by Address Resolution Protocol (ARP) and Internet Control Message Protocol (ICMP). Although it has a new name, ND is actually just a set of complementary ICMP messages that allow IPv6 nodes on the same link to discover link layer addresses and to obtain and advertise various network parameters and reachability information. In a typical scenario, a host starts the process of autoconfiguration by self-configuring a link-local address to use temporarily. This address can be formed by adding a generic local address prefix to a unique token (typically the host's IEEE LAN interface address). Once

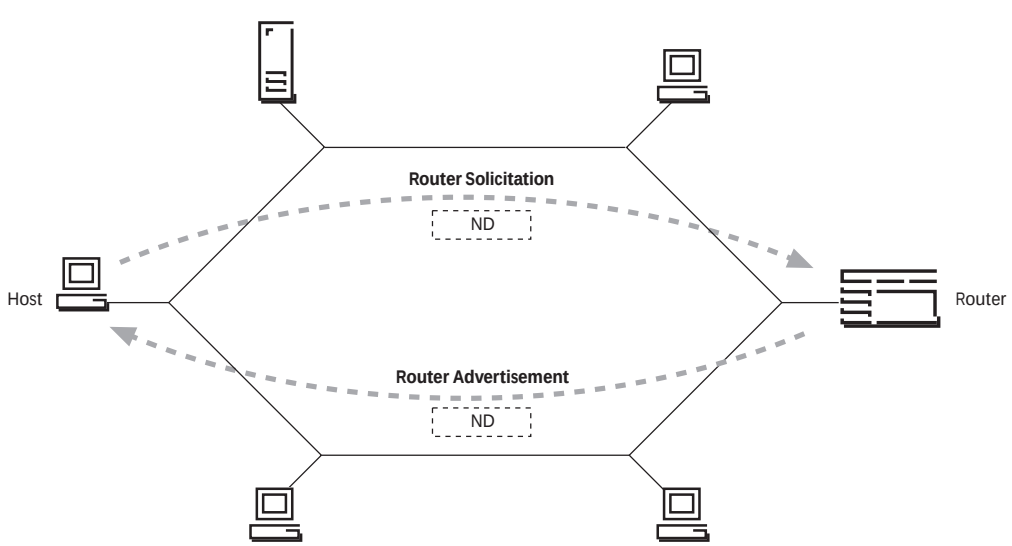
this address is formed, the host sends out an ND message to the address, to ensure that it is unique. If no ICMP message comes back, the address is unique. If a message comes back indicating that the link-local address is already in use, then a different token is used (e.g., an administrative token or a randomly generated token).

Using the new link local address as a source address, the host then sends out an ND router solicitation request. The solicitation is sent out using the IPv6 multicast service. Unlike the broadcasted ARPs of IPv4, IPv6 ND multicast solicitations are not necessarily processed by all nodes on the link, which can conserve processing resources in hosts. (IPv6 currently defines several permanent multicast groups for finding resources on the local node or link, including an all-routers group, an all-hosts group, and a DHCP server group). Routers respond to the solicitation messages from hosts with a unicast router advertisement that contains, among other things, prefix information that indicates a valid range of addresses for the subnet. Routers also send these advertisements out periodically to local multicast groups, whether or not they receive solicitations. ND message exchange is shown in Figure 16.

Using the router advertisement message, the router can control whether hosts use stateless or stateful autoconfiguration methods. In the case of stateful autoconfiguration, the host will contact a DHCP or similar address server, which will assign an address from a manually administered list. DHCP is increasingly popular for autoconfiguration in IPv4 networks and the standard is being extended to the IPv6 environment.

With the stateless approach, a host can automatically configure its own IPv6 address without the help of a stateful address server or any human intervention. The host uses the globally valid address

Figure 16 | **ND Message Exchange**

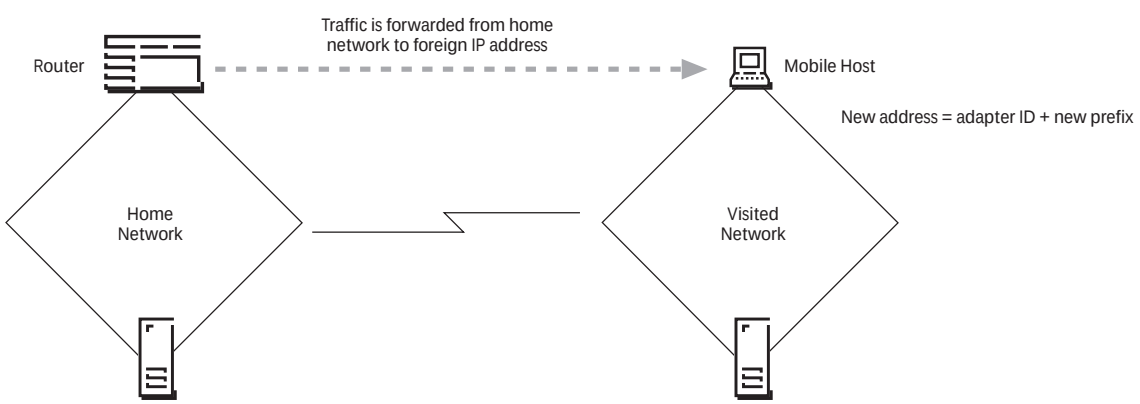


prefix information in the router advertisement message to create its own IPv6 address. This process involves the concatenation of a valid prefix with the host's link layer address or a similar unique token. As long as the token is unique and the prefix received from the router is correct, the newly configured IP address should provide reachability for the host that extends to the entire enterprise and the Internet at large.

The advantages of stateless autoconfiguration are many. For instance, if an enterprise changes service providers, the prefix information from the new provider can be propagated to routers throughout the enterprise, and hence to all stateless autoconfiguring hosts. Hypothetically, if all hosts in the enterprise use IPv6 stateless autoconfiguration, the entire enterprise could be renumbered without the manual configuration of a single host. At a more modest level, workgroups with substantial move/change activity also benefit from stateless autoconfiguration because hosts can receive a freshly configured and valid IP number each time they connect and reconnect to the network.

To support the growing universe of mobile computing devices, IETF workers have formulated a draft plan to allow IPv6 hosts to maintain connectivity with their "home" IP address while on the road. Before leaving on a trip, users will be able to request that their local router forward all traffic destined for their home IP address to a temporary "foreign" address (see Figure 17). The foreign address is typically autoconfigured by concatenating the mobile host's token (e.g., a LAN adapter address) with the prefix of the foreign network. At each stop on the trip, a new prefix can be used. This approach reduces the complication involved when name servers try to resolve names to addresses of mobile computers that are often not at their home network. With the IP forwarding features, DNS entries can remain essentially untouched, even if a host moves to the other side of the world and all points in between.

Figure 17 | **Forwarded IP Traffic**



To further facilitate host renumbering in highly dynamic situations, IPv6 has a built-in mechanism to create a graceful transition from old to new addresses. Fundamental to this mechanism is the ability of IPv6 nodes to support multiple addresses per interface. IPv6 addresses assigned to an interface can be identified as valid, depreciated, or invalid. In the renumbering process, an interface's address would become depreciated when a new address was automatically assigned (e.g., in the case of network renumbering). For a period of time after the new (valid) address is configured, the depreciated address continues to send and receive traffic. This allows sessions and communications based on the older address to be finished gracefully. Eventually the depreciated address becomes invalid and the valid address is used exclusively. Multiple IP addresses allow renumbering to occur in a highly dynamic, nondisruptive manner that is virtually transparent to end users and applications.

The above described stateless autoconfiguration process is particularly suited to conventional IP/LAN environments with 48-bit addressing and native multicast services. Other network environments with different link characteristics may require modified or alternative configuration techniques. For instance, current ATM networks do not inherently support multicast services or 48-bit IEEE addressing, due to the use of virtual circuits and telephony-style calling numbers. Multicasting solutions for ATM are seen in the emerging Multicast Address Resolution Server (MARS) that is being developed for IPv4 multicast over ATM. Plans are being devised to use MARS-style functionality to extend the IPv6 Neighbor Discovery protocol across ATM networks. This would allow network renumbering and stateless autoconfiguration to take place seamlessly in hybrid ATM/IPv6 fabrics.

Other Protocols and Services

The preceding discussion focuses on some of the more innovative and radical changes that IPv6 brings to internetworking. In many other areas, protocols and services will operate much the same as they do in the current IPv4 regime. As the industry moves to IPv6, DHCP and DNS servers are being modified to accommodate 128-bit addresses, but in terms of basic functionality, there will be little change. This is also generally true for interior and exterior routing protocols.

Open Shortest Path First (OSPF) protocol, the cornerstone of high-performance, standards-based internetworking, is the IETF recommended Interior Gateway Protocol (IGP) for IPv6. OSPF is being updated with full support for IPv6, allowing routers to be addressed with 128-bit addresses. The 32-bit link-state records of current OSFP will be replaced by 128-bit records. In general, the OSPF IPv6 link-state database of backbone routers will run in parallel with the database for IPv4 topologies. In this sense, the two versions of OSPF will operate as "ships in the night," just as the routing engines for NetWare, DECnet, AppleTalk, and other protocols coexist in the same router without major interaction. Given the limited nature of the OSPF IPv6 upgrade, those engineers and administrators who are proficient in OSPF for IPv4 should have no problems adapting to the new version. An updated version of RIP is also available, referred to as RIPng.

As with the interior gateway protocols, work is underway to create IPv6-compatible versions of the exterior gateway protocols that are used by routers to establish reachability across the Internet backbone between large enterprises, providers, and other autonomous systems. Today's backbone routers use the Border Gateway Protocol

(BGP) to distribute CIDR-based routing information throughout the Internet. BGP is known by providers and enterprises and has a large installed base. Consequently, BGP has the inside track for IPv6. Currently, work is underway to define BGP extensions that will allow it to be used to exchange reachability information based on the new IPv6 hierarchical address space.

Transition Scenarios

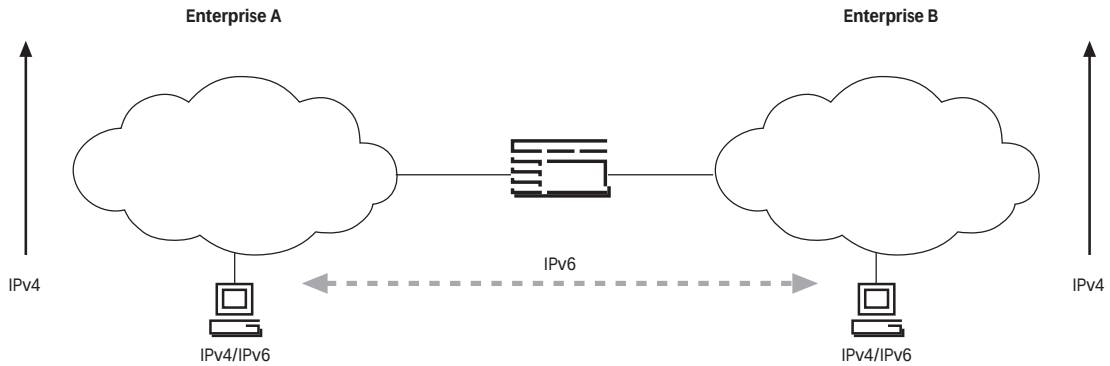
Part I of this paper provided an overview of the major transition mechanisms that are integral to the IPv6 design effort. These techniques include dual-stack IPv4 /IPv6 hosts and routers, tunnelling of IPv6 via IPv4, and a number of IPv6 services, including IPv6 DNS, DHCP, MIBs, and so on. The flexibility and usefulness of the IPv6 transition mechanisms are best gauged through scenarios that address real-world networking requirements.

Scenario 1: No Need to NAT Take, for instance, the case of two large, network-dependent organizations that must interface operations due to a merger and acquisition (M&A), or a new business partnership. Both of the enterprises in this scenario have large IPv4-based networks that have grown from small beginnings. Both of the original enterprises have a substantial number of private IPv4 addresses that are not necessarily unique within the current global IPv4 address space. Combining these two non-unique address spaces could require costly renumbering and restructuring of routers, host addresses, domains, areas, exterior routing protocols, and so on. This scenario is quite common in the current business climate, not only for M&A projects, but also for large outsourcing and customer/supplier networking relationships, where many hosts from the parent, outsourcer, supplier, or partner must be integrated into an existing enterprise address structure. Regardless of the scenario, IPv6 is an excellent approach to this challenge.

The task of logically merging two enterprise networks into a single autonomous domain is an expensive and potentially disruptive project. To avoid the cost and disruption of comprehensive renumbering, enterprises may be tempted to opt for the stopgap solution of a network address translator (NAT). In the case of the M&A scenario, a NAT could allow the two enterprises to maintain their private addresses in a more or less *status quo* fashion. To accomplish this, a NAT must conduct address translation in real time for all packets that move between the two organizations. Unfortunately, this solution introduces all the problems associated with NATs that were discussed in Part I, including performance bottlenecks, lack of scalability, lack of standards, and lack of universal connectivity among all the nodes in the new enterprise and the Internet.

In contrast with NAT, IPv6 provides a robust "future-oriented" solution to the logical integration of two physical networks (see Figure 18). For the sake of the discussion, the two originally independent enterprises will be known as Enterprise A and Enterprise B. The first step is to determine which hosts need access to both sides of the new organization. These hosts are outfitted with dual IPv4/IPv6 stacks, which allow them to maintain connectivity to their original IPv4 network while also participating in a new IPv6 logical network that will be created "on top" of the existing IPv4 physical infrastructure.

Figure 18 | **IPv6 Unites Private Address Spaces**



It's likely that the accounting department of the integrated enterprises will have financial applications on servers that will need to be accessed by accounting employees in both Enterprise A and Enterprise B. Both servers and clients will be given IPv6, but they will also retain their IPv4 stack components. The IPv6 sessions of the accounting department will travel over the existing local and remote links as "just another protocol," requiring no changes to the physical network. The only requirement for IPv6 connectivity is that routers that are adjacent to accounting department users must be upgraded to IPv6 capabilities. Where end-to-end IPv6 connectivity can't be achieved, one of the IPv4/IPv6 tunnelling techniques can be employed.

As integration continues, other departments in the newly merged enterprises will also be given IPv4/IPv6 hosts. As new departments and workgroups are added, they may be given dual-stack hosts, or in some cases, IPv6-only hosts. Hosts that require communications to the outside world via the Internet will likely receive dual stacks to maintain compatibility with IPv4 nodes exterior to the enterprise. But in some cases, hosts that only require access to internal servers and specific outside partners may be able to achieve connectivity with IPv6-only hosts. A migration to IPv6 presents the opportunity for a fresh start in terms of address allocation and routing protocol structure. IPv6 hosts and routers can immediately take advantage of IPv6 features such as stateless autoconfiguration, encryption, authentication, and so on.

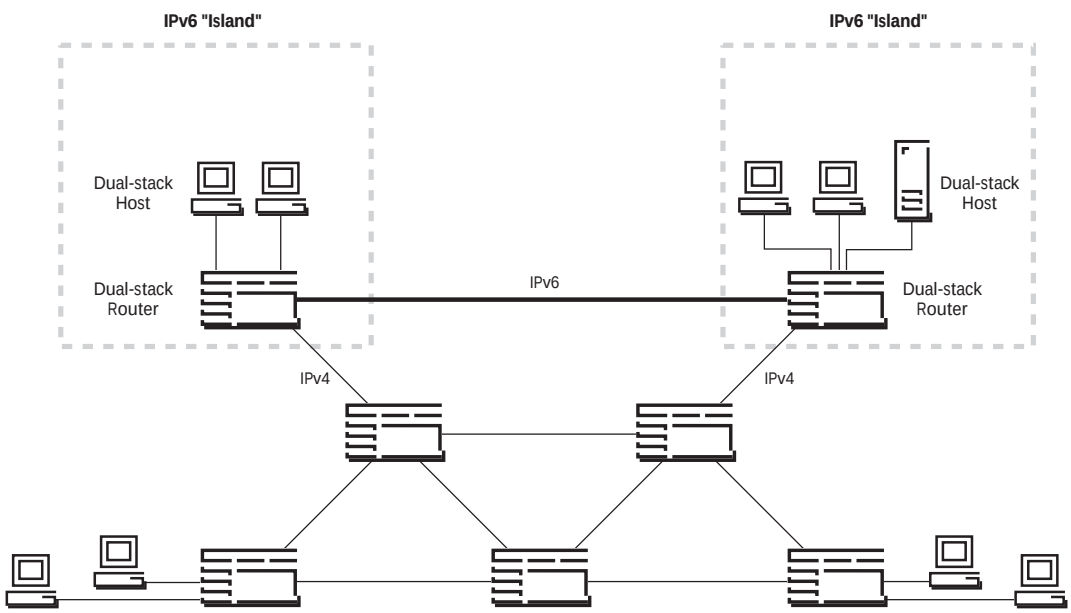
Scenario 2: IPv6 from the Edges to the Core

For a great many corporate users, connectivity requirements focus primarily on access to local e-mail, database, and applications servers. In this case, it may be best to initially upgrade only isolated workgroups and departments to IPv6, with backbone router upgrades implemented at a slower rate. IPv6 protocol development is more complete for "edge" routing than for high-level backbone routing, so this is an excellent way for enterprises to gracefully transition into IPv6. As shown in Figure 19, independent workgroups can upgrade their clients and servers to dual-stack IPv4/IPv6 hosts or IPv6-only hosts. This creates "islands" of IPv6 functionality.

As enterprise-scale routing protocols such as OSPF and BGP for IPv6 mature, the core backbone IPv6 connections can be deployed. After the first few IPv6 routers are in place, it may be desirable to connect IPv6 islands together with router-to-router tunnels. In this case, one or more routers in each island would be configured as tunnel endpoints. As described in Part I, when hosts use full IPv6 128-bit addressing, tunnels are manually configured so that the routers participating in tunnels know the address of the endpoints of the tunnel. With IPv4-compatible IPv6 addresses, automatic, nonconfigured tunnelling is possible.

From a routing protocol standpoint, tunnels appear as a single IPv6 hop, even if the tunnel is comprised of many IPv4 hops across a number of different media. IPv6 routers running OSPF can propagate link-state reachability advertisements through tunnels, just as they would across conventional point-to-point links. In the IPv6 environment, OSPF will have the advantage of flexible metrics for tunnel routes, to ensure that each tunnel is given its proper weight within the topology. In general, routers make packet-forwarding decisions in the tunnelling environment in the same way that they make decisions in the IPv6-only network. The underlying IPv4 connections are essentially transparent to IPv6 routing protocols.

Figure 19 | **Islands of IPv6**



Bay Networks Strives for IPv6 Leadership

Adaptive Networking, Bay Networks strategy for transforming today's networks into the IP-optimized networks of tomorrow, is predicated on four cornerstone technologies: routing switches, management, access, and IP services. Bay Networks support for IPv6 is an important element of its IP services strategy, and is comprised of Enabling Services, Application Services, and Integration Services.

IPv6 features such as encryption, tunneling, stateless autoconfiguration, and others (outlined in the body of this paper) fall into the Enabling Services category, allowing service providers to offer new and value-add services while enhancing the overall quality of network usage for enterprise customers. And Bay Networks implementation of IPv6-based migration strategies, such as dual stack and tunneling, falls into the Integration Services category by enabling a smooth transition to the new protocol.

With support for IPv6, Bay Networks strengthens its leadership in the area of IP services and moves customers closer to their overall goal of optimizing their networks for IP. IPv6 is a technology that Bay Networks sees as highly complementary to its industry-leading line of routing, switching, remote access, and network management products. Bay Networks is an active member of the IETF IP Next Generation Working Group which supports the effort to finalize the design of IPv6 and bring the standard to life on production network devices.

Some key IPv6-related standards work that has been authored or co-authored by Bay Networks engineers:

Multiprotocol Extensions for BGP, IETF Draft, D. Haskin, J. Stewart (ISI)

IP Version 6 over PPP, IETF RFC2023, D.Haskin, E.Allen

Management Information Base For IP Version 6: Textual Conventions And General Group, IETF Draft, D. Haskin, S. Onishi

Management Information Base For IP Version 6: ICMPv6 Group, IETF Draft, D. Haskin, S. Onishi

Management Information Base For IP Version 6: UDP and TCP Group, IETF Draft, D. Haskin, S. Onishi

RIPng for IPv6, IETF RFC2080, G.Malkin, R.Minnear (Coauthors)

Routing Aspects Of IPv6 Transition, IETF Draft, R. Callon, D. Haskin

For more sales and product information, please call **1-800-8-BAYNET**.

United States

Bay Networks, Inc.
4401 Great America Parkway
Santa Clara, CA 95054
1-800-8-BAYNET

Bay Networks, Inc.
8 Federal Street
Billerica, MA 01821-5501
1-800-8-BAYNET

Europe, Middle East, and Africa

Bay Networks EMEA, S.A.
Les Cyclades – Immeuble Naxos
25 Allée Pierre Ziller
06560 Valbonne, France
+33-4-92-96-69-96 Fax
+33-4-92-96-69-66 Phone

Pacific Rim, Canada, and Latin America

Australia +61-2-9927-8888
Brazil +55-11-247-1244
Canada 416-733-8348
China +8610-6238-5177
Hong Kong +852-2-539-1388
India +91-11-613-7401
Japan +81-3-5402-7001
Mexico +52-5-480-1241
Singapore +65-323-3522

World Wide Web: **http://www.baynetworks.com**

Copyright © 1997 Bay Networks, Inc. All rights reserved. Bay Networks is a registered trademark and the Bay Networks logo is a trademark of Bay Networks, Inc. All other brand and product names are trademarks or registered trademarks of their respective holders. Information in this document is subject to change without notice. Bay Networks, Inc. assumes no responsibility for errors that may appear in this document. Printed in USA.

7/97 WP510-2789EC-A

Printed on recycled paper.