Juniper NETWORKS

DEPLOYING IPv6: ISSUES AND STRATEGIES

An overview of the motivations for and challenges of IPv6 deployment, and strategies for beginning a reliable, cost effective IPv6 deployment

TABLE OF CONTENTS

EXECUTIVE SUMMARY 1

INTRODUCTION 1

THE DRIVER FOR IPv6 2

IPV6 DEPLOYMENT AROUND THE WORLD 5

Japan and South Korea 5
China and India .6
United States. 6
Europe 7
Developing Nations 8

PLANNING FOR IPv6 8

Design 8
Inventory 9
Methodology 9
Milestones 10
Vendor Evaluation and Selection 11
Design and Interoperability Testing 11
Training 12
Cost and Risk Analysis 12
Project Executables 13

IMPLEMENTATION MECHANISMS 13

Dual Stacks 14
Manually Configured Tunnels 16
Automatic Tunnels 17
Translators 20
Recent Developments in Implementation Mechanisms 23

CONCLUSION. 24

FOR MORE INFORMATION 24

ABOUT JUNIPER NETWORKS 24

Executive Summary

IPv6 is coming, whether we like it or not. It isn't a matter of new features or "killer applications," although those may come with time. Rather, it is the rapid depletion of the remaining IPv4 addresses that is leaving IPv6 as the only feasible alternative for the continued growth of networks beyond the next few years. Governments and service providers in many regions of the world have been cognizant of this fact for years, and are currently in various stages of planning for IP6 deployment in their networks.

With thorough, clear planning IPv6 can be deployed safely and within acceptable costs. Understanding the elements of a good deployment plan is essential, however, as is an understanding of the various mechanisms and methodologies available for IPv6 implementation. Juniper Networks is the acknowledged leader in high-performance, low-risk IPv6 deployment, with a rich set of IPv6 features available in all JUNOS-based platforms.

Introduction

IPv6 is receiving escalating attention within the networking industry. Where only a few years ago there was widespread doubt as to whether IPv6 would ever be adopted, the meetings of network operators forums such as the North American Network Operators' Group (NANOG), the Asia Pacific Regional Internet Conference on Operational Technologies (APRICOT) and Réseaux IP Européens (RIPE) now devote substantial portions of their agendas to discussions of how to best implement the new protocol. Where a few years ago resistance to IPv6 centered on the lack of a business case, organizations worldwide are now devoting significant financial and engineering resources to IPv6 planning. And where a few years ago even those who advocated IPv6 were casual about transition timelines, there is now a growing sense of urgency around its deployment.

Most of the standards that comprise the IPv6 protocol suite have been around for since the mid 1990s. What, then, is behind the suddenly intense interest in its deployment and the growing stress on deploying sooner rather than later? Is this interest justified, and should you also be thinking about deployment? How do you determine whether IPv6 is important for your own network?

If you conclude that you should be concerned, how do you begin planning an IPv6 deployment? What factors and considerations comprise a deployment project? How do you identify – and avoid – pitfalls?

This paper begins by examining the current drivers for IPv6: The answer to why people are suddenly excited – or concerned – about IPv6. An overview of IPv6 deployment status around the world is then provided.

With that foundation, the value of a well-considered deployment plan and the elements of such a plan are considered. Finally, the paper examines the major mechanisms, tools, and approaches available for deploying IPv6 in accordance with the needs of your network and your goals.

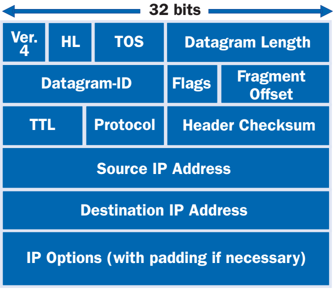
The Driver for IPv6

IPv4 was created in the 1970's, well before the advent of the world wide web, home computers, and the Internet as we know it today. In that decade no one could foresee that the protocol's 32-bit address space, representing approximately 4.3 billion addresses, could possibly be too small for what was, at the time, just an experiment.

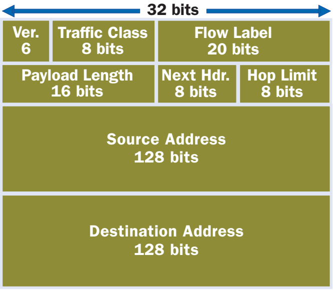
But as early as 1992 there was concern about the rapid depletion of what seemed in the 70s to be an enormous number of addresses. Much of this had to do with the way IPv4 addresses were categorized by prefixes into Class A, Class B, and Class C. Class A prefixes were 8 bits and supported 16,777,216 addresses each; Class B prefixes were 16 bits and supported 65,536 addresses each; Class C prefixes were 24 bits and supported 256 addresses each. The difficulty in the early 1990s was the large difference between Class B and Class C. Class C prefixes could only support small networks, so a great many Class B prefixes were being assigned even though most of the 65 thousand addresses within it were going to waste. As the sidebar shows, the wasteful allocation of Class B prefixes was expected to entirely deplete those addresses by 1995.

Beyond Class B address exhaustion, the rapidly rising popularity of IP networking enabled many to recognize by those years the eventual depletion of all IPv4 addresses. A new version of the protocol supporting a much larger pool of available addresses was needed. After considering a number of proposals, IPv6 was adopted.

Everyone involved in this move understood that it would take many years to develop the IPv6 standards and migrate to that new version, and that the IPv4 address space would run out well before that process could be completed. Therefore short-term solutions were needed to slow the rate at which IPv4 addresses were being handed out.



IPv4 header



IPv6 header

Four solutions, intended to work together to slow IPv4 address exhaustion, were adopted in the early 1990s:

- Classless Inter-Domain Routing (CIDR) did away with the wasteful IPv4 Class A, B, and C structure and allowed allocation of prefixes according to what fit actual needs. So if a network operator needed an address space twice as big as the 24-bit Class C, but substantially smaller than the 16-bit Class B, he could be assigned a 23-bit prefix.
- Dynamic Host Configuration Protocol (DHCP) enabled the dynamic assignment of IP addresses from a shared pool to network devices. Working on the premise that not all devices would be on-line at the same time, a small number of addresses could serve a relatively large number of devices.
- Private IP Addresses reserved a block of IPv4 addresses for use in networks that did not need to communicate outside of a private network, allowing the same addresses to be reused in many different networks.
- Network Address Translation (NAT) allowed a large number of privately addressed devices to be represented to the Internet by a small pool of public addresses.

Additionally, the Internet Assigned Numbers Authority (IANA) enacted several new rules for the assignment of new IPv4 address prefixes:

- Networks who had been assigned address space before CIDR was adopted in November of 1993 could not receive new allocations until they had proven that they had used up most of their previous assigned addresses.
- Networks asking for new address allocations had to justify their need for the addresses, both in terms of public communication needs and number of devices to be supported.
- Networks qualifying for new IPv4 address allocations would be given “just enough” for immediate and projected near-term needs, and would have to show that those addresses had been efficiently used before they could qualify for more.

These short-term solutions and new rules worked exceedingly well throughout the 1990s. But by 2000, an explosion of new address demands renewed the pressure on the IPv4 address space. These demands were (and still are) created by several factors:

- An explosion of Internet applications, games, information sources, and business transactions
- The movement of traditional services such as voice and video from legacy circuit-based infrastructures to IP networks
- Millions of new IP-enabled mobile handsets, with millions more projected in the near future
- Expanding economies in populous countries such as China and India, and developing economies throughout the world
- Burgeoning consumer electronics industries finding new ways to exploit IP capabilities
- Emerging IP-enabled sensor networks for industrial, medical, and military applications

China had 220 million Internet users in February 2008, according to the China Internet Network Information Center, surpassing the United States' 216 million users. The Economist magazine, in its September 6 – 12 issue, states that 29% of these users access the Internet using their mobile phones. And according to China's Ministry of Industry and Information Technology, there were 601 million Chinese mobile phone users in June of 2008.

These combined dynamics have produced, in the eight years of this century, an accelerating depletion of the remaining IPv4 address space; as of October 2008, only 15% of the entire IPv4 address space remains for allocation. Several authoritative studies on the rate of IPv4 address allocation convincingly conclude that the IANA's pool of available addresses will run dry in late 2010, with the Regional Internet Registries (RIRs) depleting approximately one year after that.[1]

Internet service providers and government agencies worldwide have taken note of these facts, and have begun to act with increasing aggressiveness to deploy IPv6 before the clock runs out. Agencies planning new IP networks and infrastructures in the near future also recognize the importance of IPv6 capability; as the IANA and RIRs tighten their IPv4 allocation policies in a push to adopt IPv6, new networks will find that soon the only addresses available to them are

[1] The most informative of these studies is found at www.potaroo.net/tools/ipv4/. This site continuously monitors IPv4 allocations and updates its projections of the address pool exhaustion.

IPv6. Enterprises such as Google also understand that their services must be accessible to IPv6 Internet users, and are active in upgrading their public servers.

IPv6 has many advantages over IPv4: improved mobility, potentially better multicast capabilities, easier extensibility, more efficient packet processing, and cleaner security capabilities. But none of these are important enough to drive a transition to IPv6 alone. The real driver for IPv6 in this first decade of the 21st century is the same one that drove its development in the last century of the 20th century: Enough addresses to support the continued growth of the Internet and IP services into the foreseeable future.

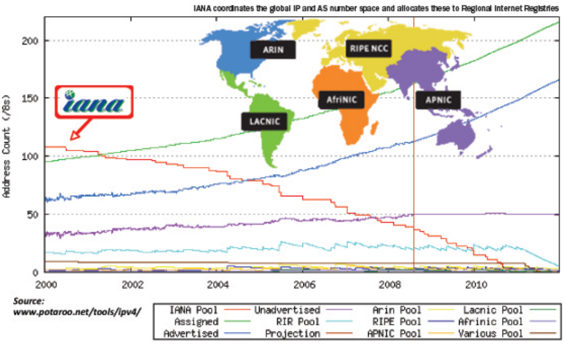
IPv6 Deployment Around the World

One of the first questions asked by almost anyone considering IPv6, anywhere in the world, is, “What is the rest of the world doing?” An examination of IPv6 activities in various regions is instructive of the motivations for deploying IPv6 and the progress that has been made in moving toward an IPv6 Internet.

Japan and South Korea

Japan was the first country to move forward with a concerted, government-supported IPv6 initiative (the e-Japan Initiative). Innovative research continues to be conducted by Japanese organizations such as the WIDE Project; a multitude of IPv6 protocol stacks for operating systems, IPv6 enabled systems, and IPv6 applications have come out of Japan. TAHI is one of the most widely used IPv6 conformance and interoperability test suites. The IPv6 Ready logo program is managed from Japan. And NTT/Verio is far in advance of other telecoms in the deployment of IPv6.

The End of the Road Comes into View



Source:
www.potaroo.net/tools/ipv4/

The driver for Japan's early enthusiasm was and continues to be the consumer electronics industries on which represents such a large portion of the Japanese economy. The makers of everything from game systems to mobile handsets to cameras understood the value of having their products connect to the Internet, and understood too that IPv4 did not have the address capacity to support the numbers of network-enabled devices they envisioned. IPv6 was recognized by the Japanese electronics industry, renowned for looking well beyond the next two or three years, as vital to its continued growth and innovation.

South Korea, whose economy was also powered by huge consumer electronics manufacturers, was not far behind Japan in its push for IPv6 adoption. Like Japan, the South Korean government provided leadership and financial incentives to early adopters. Taiwan, while not as far along as Japan and South Korea, is also motivated to support IPv6 because of its electronic industries.

China and India

China has a government-led and funded IPv6 deployment mandate called the China Next-Generation Internet (CNGI) Project. And while China, like Japan and Korea, has a burgeoning electronics industry, their motivation for IPv6 comes more from the size of their population and their dynamic economy. As wealth rises in China, more and more people are getting online both with PCs and with mobile devices. At the end of 2008, there are approximately 654 million IPv4 addresses remaining; there are 1.3 billion people in the People's Republic of China. There are not enough IPv4 addresses left to give even one address to every Chinese citizen. IPv6 is the only way to bring the Internet to the Chinese population.

China highlighted its progress with IPv6 at the 2008 Olympics. Lighting control systems and security cameras throughout the Olympic venues operated over IPv6, and IPv6-enabled sensors in taxis helped ease traffic congestion.

Close behind China in population size is India. And while the Indian economy is not yet expanding as fast as China's, and has begun its expansion more recently, it is growing. And while IPv6 deployment is not yet being as aggressively pushed as it is in China, the motivations for IPv6 in India are the same as China's and will soon be on the rise.

United States

While governments in Japan, South Korea, and China have spurred IPv6 deployment through direct initiatives and funding, the United States government is pushing IPv6 though a different means. Rather than issuing policy directions to service providers and network equipment vendors, it has issued mandates that government agencies themselves will adopt IPv6 and have made IPv6 support a requirement for selling IP equipment and services to the government. These mandates began with the Department of Defense in 2003 and then spread to other agencies through directives from the Office of Management and Budget. Because the agencies of the US government collectively represent an enormous customer base, service providers and vendors are scrambling to meet federal IPv6 guidelines in order to protect existing business.

Unlike Asian countries, the US government mandates are not primarily based on anticipated IP address shortages. The Department of Defense, for example, has a huge reserve of IPv4 addresses. Instead, several expected improvements in the protocol are driving federal interest such as superior IPv6 mobile capabilities, better multicast features, and IPv6 plug-and-play addressing that will greatly improve peer-to-peer network models and make mobile ad-hoc networks practical.

The government is not alone in driving IPv6 adoption in the United States, however. Most of the world's Tier 1 service providers are US-based, such as AT&T, Level 3, Global Crossing, Sprint, Qwest, and Verizon Business. These providers, forming much of the "core" of the Internet, are looking closely at the IPv4 depletion rates and understand the need to deploy IPv6 in order to continue expanding their business. All of them, accordingly, are either actively deploying IPv6 or intend to begin deployment projects in the near future.

US Internet application providers are also preparing for IPv6. For example, Google is currently implementing IPv6 to insure that their services are ready for the growing number of IPv6 Internet users.

Europe

More IPv6 address allocations have been made to Europe than to any other region of the world. Most of this has to do with the number of individual European countries active in the IPv6 arena, compared to the number of countries in other regions of the world pursuing IPv6. And while there are numerous research and development projects happening throughout European countries, there are also common motivations for IPv6 that can be attributed to that region.

A major driver for IPv6 in Europe is mobile telephony and European telcos' strong investments in 3G technology. Leadership in the adoption of IPv6 has similarities to the government leadership in some Asian countries: The European Union staunchly supports IPv6 as a vehicle toward competitive growth and is funding over 30 research and development projects throughout its member countries. With its i2010 initiative, the EU plans for 25% of European Internet users to access the Internet and their most important content via IPv6 by 2010. The EU is also focusing on the top 100 European websites, encouraging them to become IPv6 accessible.

The EU also has a strategy similar to that of the US government, promoting the adoption of IPv6 by encouraging its member states to include the protocol in their own network purchasing requirements. The EU is also making IPv6 a requirement for its own networks.

Developing Nations

With new IPv4 addresses expected to become unavailable around 2011 – 2012, IPv6 is particularly important to developing nations who see demands for IP networks within their borders at or after that time. Internet access in these countries is expected to be primarily through mobile handsets rather than traditional fixed, PC-based networks. Therefore the issues of mobility, and the means by which IPv6 provides superior capabilities to mobile networks, are of great interest in such economies.

Planning for IPv6

Creating a successful IPv6 implementation plan is in most ways no different from planning for the implementation of any new technology. A few overarching rules apply:

- Deploy the technology incrementally
- Back up your design assumptions with practical testing
- Establish sensible, liberal timelines

There are, however, some factors that make an IPv6 implementation plan unique. Most of these involve the specifics of IPv6 and its implementation mechanisms as discussed in a subsequent section. Planning for IPv6 must also take into account the relative lack of extensive experience with the protocol and the resulting dearth of IPv6 deployment best practices. New technologies increase project risk, but careful planning can bring those risks back to an acceptable level.

The following subsections describe the components of an IPv6 implementation plan that help you control risk and costs and insure a successful completion.

Design

While the implementation plan in general describes how you are going to accomplish the introduction and normalization of IPv6 into your network, the design starts it off with a description of what you plan to accomplish. Certainly the design is a description of what the network is to look like upon project completion, just as a structural design shows what a building will look like at the end of a construction project. But it is more than that. In providing the vision of the end results, it also provides the objectives of the project: what you plan to accomplish.

The logical extension of what you plan to accomplish is why you want to accomplish it. In other words, the technical objectives of the implementation project. So the design, describing the project terminus, is the first element of the project plan. You shouldn't embark on a journey before knowing exactly why you are going and what rewards you expect to be awaiting you at the end.

The design serves another essential prerequisite to the start of the project: Its clearly stated objectives are the foundation of the business case you must make to gain funding for the plan. Not the funding for the project itself – that comes at the end of the implementation plan – but the funding for the personnel, equipment, and time required to develop the plan.

Inventory

A thorough inventory of your network is an essential first step to implementation planning: You cannot efficiently make changes if you do not know what must be changed.

The network inventory must cover everything that IPv6 will touch: Routers, servers, and hosts; the operating system versions they run; security systems; management systems; and backoffice systems. User applications must also be inventoried. The inventory must provide a clear listing of what already supports IPv6, what must be upgraded, and what must be replaced.

Methodology

There are three approaches for deploying IPv6 in a network:

- *Core to Edge:* IPv6 is implemented first in the routers forming the core of the network, usually using dual stacked interfaces, and progressively expanded toward the edge of the network. This methodology has the advantage of implementing first where it is easiest, as most core router software either already supports IPv6 or can support it with a simple upgrade. This gains you more time to address the more difficult security and management implementations as the core is being converted. Core to edge also tends to be the safest approach, allowing operations and engineering personnel time to become acquainted with the protocol before it reaches the users.

- *Edge to Core:* IPv6 is implemented first at the edge of the network and then expanded toward the core. Manual tunnels such as GRE or MPLS are used to connect edge devices across the core during the interim. This approach is advantageous when IPv6 must be turned up relatively quickly for a customer requiring it or when a network must otherwise demonstrate early IPv6 capability. It is also valuable when the core consists of legacy routers that either cannot support IPv6 but can support a tunneling technology or that can only be upgraded with difficulty.
- *IPv6 Islands:* Certain segments throughout the network, ranging from individual devices to complete sites, are converted. The islands can be interconnected with manual or automatic tunnels, or a combination of the two. As the implementation project progresses, the IPv6-capable islands grow until they begin to merge, and toward the end of the project there are IPv4-only islands in the midst of an IPv6-capable ocean. This approach is useful when the network's existing IPv6 capabilities are scattered or when IPv6 must be quickly added to specialized systems throughout the network.

Milestones

With a methodology selected, you can begin defining milestones marking the completion of project phases. Whichever methodology is used, incremental deployment is essential to controlling project risk. Hence at a milestone a certain phase should be completed, testing and verification should be performed, and the capabilities expected at that milestone certified before moving to the next project phase.

Collectively, the milestones comprise a project timeline. Longer timelines, when they can be supported, have multiple benefits:

- A long project timeline reduces risk by allowing sufficient time for testing and verification, and for reassessing aspects of the project that give unexpected results.
- A long project timeline can significantly reduce costs by allowing the introduction of new IPv6 capable systems within the normal network upgrade cycles. That is, most network systems are replaced or undergo major upgrades every 3 – 5 years. An IPv6 implementation project spanning those years allow you to bring IPv6 capability in during those planned changes rather than being forced to replace or upgrade a system early, at a capital loss.
- A long project timeline insures that your operations and engineering personnel are introduced to IPv6 gradually, giving them time to build expertise.

Vendor Evaluation and Selection

With a design and methodology selected, systems to be changed identified, and a timeline set, the next step in the implementation project is the evaluation and selection of vendors. This step can represent the first significant expenditure of the project planning (although the inventory can in some circumstances be expensive). A through evaluation of vendors requires lab testing to verify standards compliance.

Reliable vendor evaluation requires much more than a “Supports IPv6” checkbox on an RFP. A vendor could truthfully check off that box if its product supports nothing more than the ability to have an IPv6 address configured on an interface and a few core protocols. The previous steps in the implementation plan will have produced an exact listing of the IPv6 protocols, capabilities, and features that are required for the implementation project, and a vendor must be able to positively respond to the requirements to be accepted as a candidate for selection.

Another aspect of the evaluation must be the cost, difficulty, and risk of upgrading an operating system to gain the needed IPv6 features. The preference, of course, is to have the desired features already available in existing software. But if systems must be changed or upgraded, the processes of upgrading must be taken into consideration.

After a list of candidate vendors is established, compliance testing should be performed. Many vendors’ IPv6 implementations are immature enough that verification testing in a lab is important to insure that the implementation is complete and bug-free. Neglect of this step can result in unpleasant surprises during the deployment project.

Design and Interoperability Testing

Lab testing of the design is another risk reduction step. While it is impractical to build the entire network in the lab, building and testing strategically selected parts of the design will yield enough information to increase the confidence in the overall design.

The selected implementation methodology should also be tested. In addition to verifying that the methodology works as expected, testing provides essential “dry run” experience for the personnel who will be responsible for executing the implementation project.

Finally, interoperability of the required IPv6 features among the selected vendors or vendor candidates should be tested. Careful adherence to open protocol standards among product vendors should assure interoperability, but vendors who miss some subtle aspects of a standard – omissions or mistakes that might slip by the conformance tests – might have an implementation that does not perform seamlessly with other vendors' implementations. Thorough testing of all aspects of the design in which two or more products must interact should reveal any interoperability problems and allow time for a vendor to correct identified shortcomings before the implementation project begins.

Training

As IPv6 is deployed in the network, operations personnel must be ready to manage it. Security personnel must be ready to protect it. Engineers must be ready to troubleshoot it. Therefore a training plan must be a part of the implementation plan.

Creating a training plan has analogs to the early steps of creating an implementation plan. The design highlights *what* the training must include. An inventory of existing knowledge and skills reveals who requires training. And a training methodology details *how* the training is to be accomplished.

Training is multi-faceted, and must be planned accordingly:

- Architects and top-tier engineers need a deep understanding of the protocols themselves. Existing knowledge will vary widely in this group, so attention to individual needs is important. Self-directed study is generally more effective than structured classroom training for these people, as long as the resources and guidelines are available to them.
- Those responsible for the day-to-day operational upkeep of the network require less in-depth protocol knowledge and more hands-on skills. Vendor-specific courses are most effective for this group.

The lab built for vendor evaluation and design verification is also a valuable training resource. Not only does it allow personnel to build hands-on skills, reinforce knowledge gained in training sessions, and closely observe protocol behavior; the lab is built to the specifics of the IPv6 network design and therefore focuses learning on the network to be implemented.

Cost and Risk Analysis

The intent of planning – any planning – is to control cost and risk. The previous steps outlined for an IPv6 implementation plan provide the data required to make an accurate cost and risk analysis of project. If either or both factors exceed acceptable thresholds the timeline, methodology, vendor selection, or in some cases even the design itself can be adjusted to bring cost and/or risk down to a tolerable level.

Project Executables

Once the project plan is complete, has been adjusted for cost and risk, and has been used to fund the implementation project, the details of the project can be developed. These are the specifics required to execute the project, such as:

- Detailed project scheduling
- Individual device configurations
- Upgrade and execution scripts
- Resource allocation
- Personnel assignments
- Backout plans

Implementation Mechanisms

Several factors must be considered when planning for IPv6:

- IPv6 and IPv4 are not by themselves interoperable, so means must be found for users of the two protocols to connect.
- The incremental approach to implementation requires mechanisms – in fact, a selection of mechanisms – that can support the various project phases.
- As the "Methodology" component of the preceding "Planning for IPv6" section explains, there are multiple approaches to deployment. It is important to have the right tools for the right methodology.

There is a virtual grab bag of technologies for implementing IPv6. Over the past decade many more have been proposed, but the ones that remain have been proven in practical, real-world IPv6 deployments. As the discussions in this section show, however, most of the technologies have both positive and negative characteristics that must be considered when making the best choices for a specific project.

The implementation mechanisms can be classified into just a few categories:

- Dual stacks
- Manually configured tunnels
- Automatic tunnels
- Translators

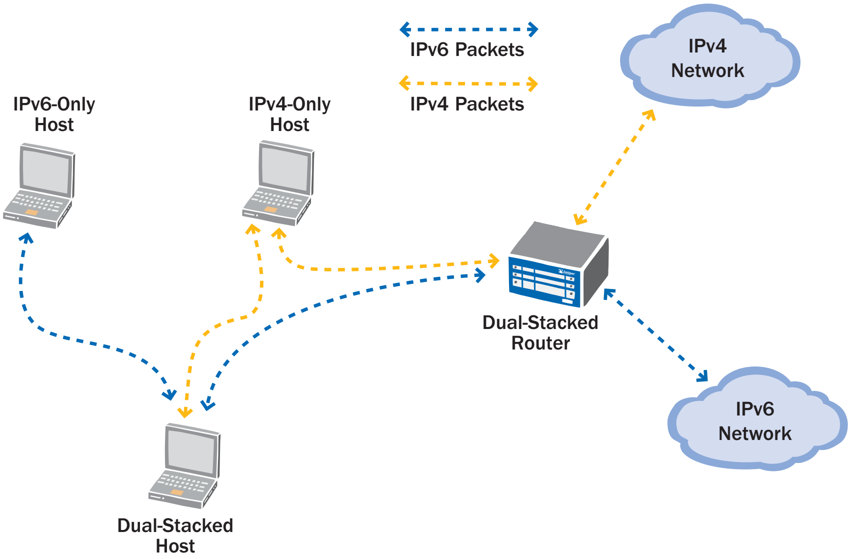
These mechanisms are listed in the order of complexity, with dual-stacking being the simplest and translators being the most complex. A sensible approach, then, is to look at the simplest mechanisms first and move to progressively more complex solutions only when the simpler ones do not meet the project requirements.

Dual Stacks

Dual stacking is the ability of a device to simultaneously support both IPv4 and IPv6 in the same interface. At the connection to the data link, the interface has both an IPv4 and an IPv6 address. At the upper protocol layers, an application can use either IPv4 or IPv6 to communicate. And other nodes can send either IPv4 or IPv6 packets to the dual stacked device. Most significantly, it means that both IPv4-only and IPv6-only devices can communicate with a dual stacked node. It is “bilingual.”

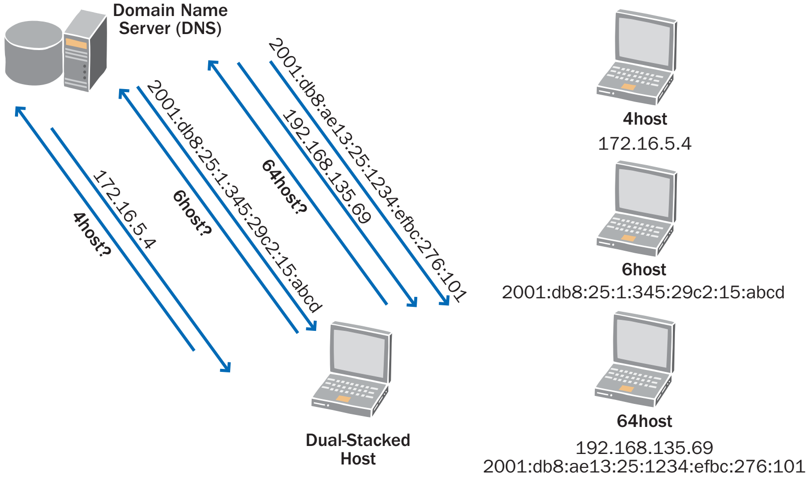
Dual stacking is best suited for core-to-edge implementation strategies. It also can be used internally in IPv6 “islands.” What both approaches have in common is that the systems on which IPv6 is deployed are directly interconnected: either in the network core or within an IPv6 island. The systems can communicate with each other using IPv6, and can communicate with the outside world – and the few systems within their topology that are still IPv4—using IPv4.

Implementation of IPv6 using dual stacks is the simplest approach because the change is driven by DNS: When a dual stacked node queries DNS for a destination and is given an IPv4 address, the node speaks IPv4 to the destination. If DNS returns an IPv6 address, the node speaks IPv6. Management of the deployment through DNS also allows exacting, incremental control. Even if a group of devices are dual stacked, they are not going to use IPv6 to communicate with other IPv6-capable devices off of their local link until the necessary records are entered in DNS.



A dual-stacked device can send and receive both IPv4 and IPv6 packets

Care must be taken, of course, that DNS does not provide an IPv6 address for a destination that cannot be reached by IPv6. For example, if DNS returns an IPv6 address to a node in site A for a destination in site B, but the two sites are separated by an IPv4-only segment, the two nodes cannot communicate.


Domain Name Server (DNS)
4host?
172.16.5.4
6host?
2001:db8:25:1:345:29c2:15:abcd
64host?
192.168.135.69
2001:db8:ae13:25:1234:efbc:276:101
4host
172.16.5.4
6host
2001:db8:25:1:345:29c2:15:abcd
64host
192.168.135.69
2001:db8:ae13:25:1234:efbc:276:101
Dual-Stacked Host

DNS can return an IPv4 address, and IPv6 address, or both in response to a query.

Complications can also arise when DNS returns both IPv4 and IPv6 addresses for a destination. If one node gives first preference to an IPv4 address and another node gives first preference to an IPv6 address, a periodic delay might occur while the nodes resolve their differences. Therefore consideration must be given, when using dual stacks, to how nodes interact in that environment.

Dual stacking is also the simplest approach to adding IPv6 support to a group of interconnected routers. They can run an integrated routing protocol that exchanges both IPv4 and IPv6 reachability information, such as IS-IS and BGP, or the routers can run separate, version-specific routing protocols such as OSPF or RIP. Such an internetwork can then provide dual, IPv4-only, and IPv6-only interfaces at its edge.

There is a limitation to the dual stack approach, however. Because every interface requires both an IPv4 address and an IPv6 address, it does not make sense in environments where IPv6 is being implemented specifically because IPv4 addresses cannot be acquired. On an Internet scale, dual stacking would have been the right approach to a complete migration to IPv6 five or more years ago, while IPv4 addresses were still plentiful. With time quickly running out on

IPv4, the advantages of dual stacks are reduced. Nevertheless, it remains a preferred method for individual networks where IPv4 address availability is anticipated but is not an immediate problem.

Manually Configured Tunnels

A tunnel is a logical construct in which data is encapsulated in a packet to be transported across a network. In an IPv6 deployment project, IPv6 packets can be encapsulated in IPv4 packets to be transported across an IPv4-only portion of the network. In latter stages of the project, IPv4 packets might be encapsulated in IPv6 packets for transport across IPv6-only parts of the network.

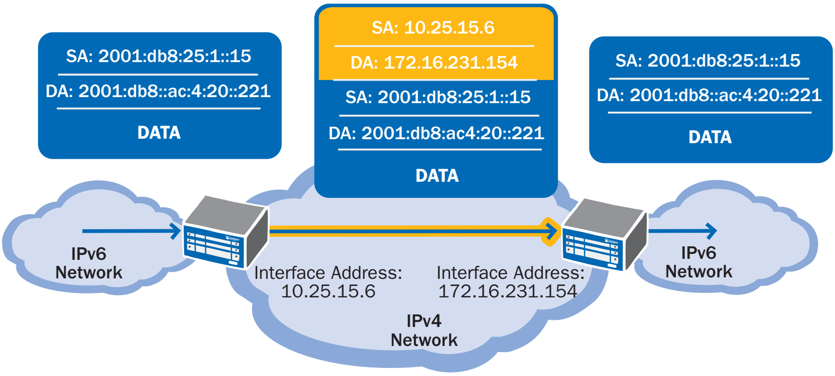
As implied by the description of transporting packets of one protocol across a part of the network that only supports the other protocol, tunnels are best suited for edge-to-core approaches and for interconnecting IPv6 "islands." Tunnels can be either manually configured or can be set up automatically; this section discusses the former, and the next section discusses the latter.

The key element for enabling a tunnel is the specification of a source and destination address for the encapsulating packet. If an IPv6 packet is encapsulated in an IPv4 packet and then transported across an IPv4 network, the IPv4 source address of the packet – the near end of the tunnel – and the IPv4 destination address of the packet – the far end of the tunnel – must be known. A manual tunnel is set up by statically configuring this information at the devices (usually routers) at each end of the tunnel.

Manual tunnels are ideal for interconnecting IPv6 sites over an IPv4 network, where the sites do not change. However, as the number of sites grows the challenge of interconnecting them with a full mesh of tunnels can become an administrative problem. The difficulty is the same as the scaling difficulty of operationally supporting a full mesh of ATM or Frame Relay virtual circuits: As the number of sites to be interconnected grows, the number of tunnels required to provide direct connections between all sites grows exponentially. Therefore manual tunnels are best used when there are a manageable number of sites to be interconnected, when inter-site communication requires only a partial mesh, or when an alternative topology such as hub-and-spoke can be used. As either the IPv6 edge expands toward the core in an edge-to-core methodology or as IPv6 islands expand and merge in the islands methodology, the tunnel scope shrinks as IPv6 boundaries meet.

Another method to create pre-established tunnels is by using MultiProtocol Label Switching (MPLS). Most large service provider networks and a few large enterprise networks operate MPLS cores, and the technology is ideal for IPv6 implementation. IPv6 can be deployed at the edge of an MPLS network without the need for deploying it in the core, which can be advantageous if the network operator would otherwise have to upgrade core routers for IPv6 support or if the operator simply wants to transport IPv6 in the same way they transport IPv4, with the same MPLS resiliency and traffic engineering capabilities.

In addition to carrying IPv6 packets natively over MPLS tunnels, MPLS can be used to build IPv6 Virtual Private Networks. For service providers, IPv6 VPNs are an ideal service for interconnecting customer sites over the provider network while providing complete privacy and separation between different customers. MPLS also allow enables the creation of point-to-point layer 2 VPNs, which appear to the customer as dedicated layer 2 links, and Virtual Private LAN Service (VPLS), in which the provider network appears to the customer sites as a single Ethernet switch. Both of these services are “layer 3 agnostic,” so IPv6 can be carried over them as easily as IPv4 or any other layer 3 protocol. By combining native IPv6 tunneling, IPv6 VPNs, layer 2 VPNs, and VPLS, all supported over the same MPLS backbone, a service provider can offer a portfolio of IPv6 solutions to fit specific customer needs.



An IPv6-in-IPv4 tunnel adds IPv6 packets in IPv4

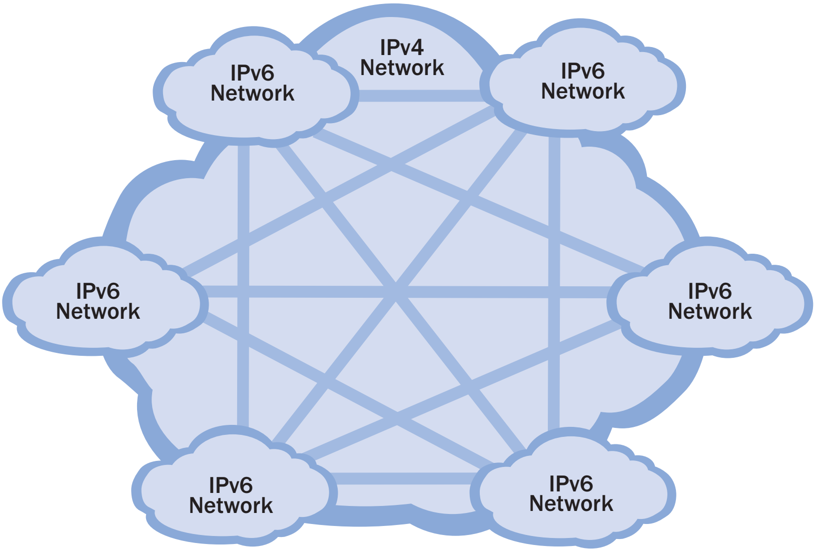
Automatic Tunnels

Automatic tunnels do away with the need to manually configure each tunnel endpoint and instead use some mechanism to automatically discover the endpoint addresses. Although there are several types of automatic tunnel – and quite a few more that were never adopted in practical deployments – they all use one of two means to discover endpoint addresses:

- An authoritative server is used to provide the endpoint addresses
- The IPv4 endpoint addresses are embedded in the IPv6 addresses of the packets to be tunneled.

The most common use of the authoritative server solution is tunnel brokers. When an IPv6 device wants to communicate with another IPv6 device over an IPv4 network, the device uses a small application to query a server. The server returns the necessary tunnel setup instructions. Public tunnel broker services are used worldwide; the best-known examples in North America are Hurricane Electric and Hexago's Freenet6.

Another server-based automatic tunneling mechanism is Microsoft's Teredo, found in Windows Vista and easily added to Windows XP. Where tunnel brokers can be used for connectivity from individual devices or from IPv6 sites, Teredo is used specifically for individual device connectivity where the device is a dual-stacked client in an IPv4 network. Like tunnel brokers, the Teredo client queries a server when it needs to communicate with another IPv6 device, and the server provides the tunnel setup information.

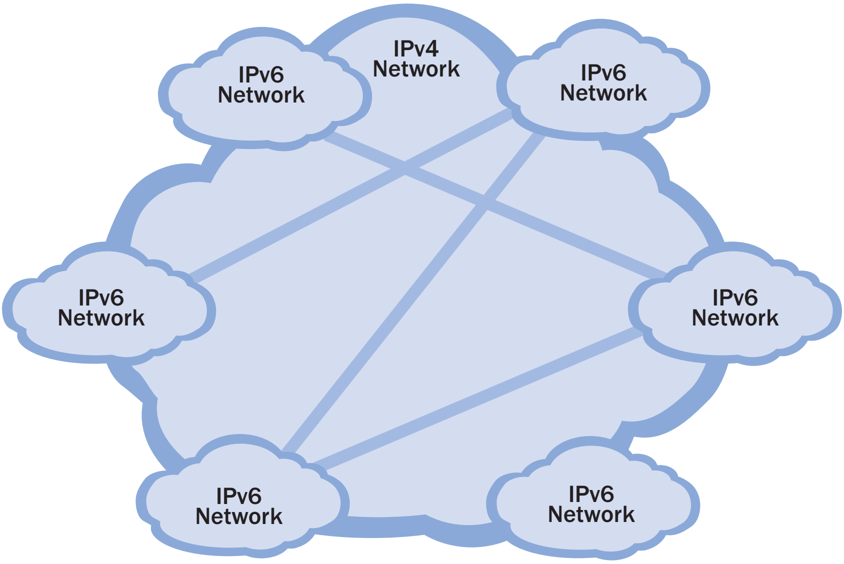


Manual tunnels offer granular control for site-to-site connections, but can pose scaling problems when full interconnectivity among many sites is required.

A disadvantage of most manual and automatic tunneling mechanisms is that they depend on publicly reachable IPv4 addresses to function properly. When a device is “hidden” behind NAT, using a private IPv4 address, NAT’s translation of the addresses in IPv4 headers prevents tunnel mechanisms that encapsulate the IPv6 packet directly behind an IPv4 header from successfully passing packets.

Teredo and most tunnel brokers have the advantage of being able to operate through IPv4 NATs by encapsulating IPv6 packets with a UDP header. By making the IPv6 packet independent of the encapsulating IPv4 addresses, and a server that can analyze the NAT, the tunnels can traverse NATs without breaking. These tunnel types, then, are ideal for implementing IPv6 on single devices or on home and small office networks where there is no public IPv4 access without passing through a NAT; IPv6 packets originated from the same devices can pass transparently through the NAT while NAT continues to translate IPv4 addresses.

6to4 is the most well-known example of an automatic tunneling mechanism that uses IPv4 addresses embedded in IPv6 addresses to determine tunnel endpoints. When two IPv6 sites are separated by an IPv4 network, a device in one site can use a specialized IPv6 prefix of 2002::/16 to create the source and destination addresses to a device in the remote site. The 32 bits following



Automatic tunnels can improve scalability by being configured only as needed and only for the duration of a session.

the 16-bit 6to4 prefix is an embedded IPv4 address: The source address of the IPv6 packet embeds the near end address of the IPv4 tunnel, and the destination address of the IPv6 packet embeds the remote end address of the tunnel. 6to4-aware gateways at the borders between the IPv6 and IPv4 networks recognize the 2002::/16 IPv6 prefix and know to look for the tunnel addresses in the next 32 bits. The near-end 6to4 gateway then encapsulates the IPv6 packet in an IPv4 packet, using the embedded addresses as the IPv4 source and destination addresses. The remote 6to4 gateway, receiving the packet, decapsulates the IPv6 packet and sends it to the destination in its locally connected IPv6 site. When the destination responds, the process works in reverse send packets back to the first site.

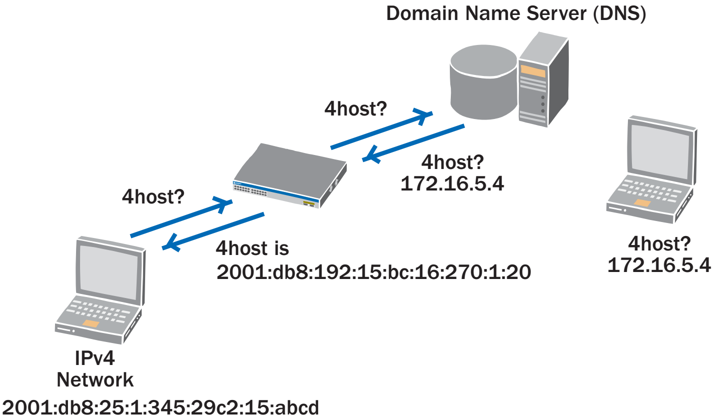
While 6to4 connects separate IPv6 sites, another automatic tunneling protocol that uses embedded IPv4 addresses – ISATAP – connects individual IPv6 devices within an IPv4 network. The principle is the same: The source and destination IPv6 addresses contain the source and destination IPv4 addresses. ISATAP is useful when there are just a few IPv6 devices in an primarily IPv4 site; it can also be useful in conjunction with 6to4, allowing an IPv6 device to tunnel via ISATAP to a 6to4 gateway at a local site, and then use 6to4 to tunnel to a remote site.

Automatic tunnels avoid the operational scaling concerns of manual tunnels by enabling a tunnel to be set up for the duration of a communication session and then torn down afterward. However, as with any automatic protocol surrendering some control in favor of dynamic functionality can introduce a few concerns. Some automatic tunnels, such as 6to4, have no authentication mechanism and are therefore open to abuse. Tunnel brokers typically use authentication and therefore reduce security concerns. Automatic tunnels can also, by their transitory nature, be difficult to troubleshoot. Errant ICMP messaging within a tunnel, in which an ICMP error message is returned to a tunnel ingress point rather than to the originating IPv6 device, can cause problems with Path MTU Discovery, an important IPv6 function.

Finally, all automatic tunneling mechanisms require the IPv6 node to run some sort of tunneling application and to be aware that it is using a tunnel. They do not solve the problem of enabling an IPv6-only device to speak to an IPv4-only device. For that, a protocol translator is required.

Translators

Unlike tunnels in which packets of one protocol are encapsulated within packets of another protocol, an IPv4/IPv6 translator completely replaces the header of one protocol with the header of another protocol. Although a number of translation mechanisms have been proposed over the years, the only one to gain usage is called Network Address Translation with Protocol Translation (NAT-PT).

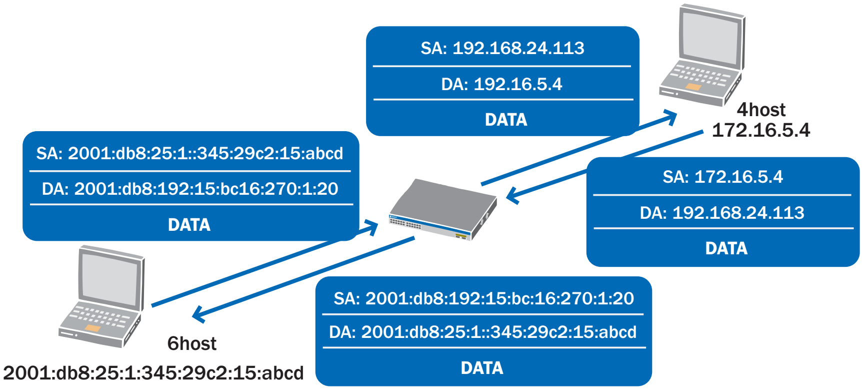


A DNS application Level Gateway (ALG) in a NAT-PT translates an IPv4 DNS response to an IPv6 DNS response.

NAT-PT maintains an assignable pool of addresses, the same way traditional NAT does. But instead of a pool of IPv4 addresses, NAT-PT's address pool is IPv6. When an IPv6-only host queries DNS for the address of a destination, the query passes through a NAT-PT. If DNS returns an IPv4 address, NAT-PT maps the IPv4 address to one of the IPv6 addresses from its pool. It then sends the DNS response on to the host, but containing the assigned IPv6 address instead of the IPv4 address DNS actually returned. The host then sends its packets to the IPv6 address; as the packets pass through the NAT-PT, the translator changes the IPv6 header to an IPv4 header. It changes the IPv6 destination address to the actual mapped IPv4 address, and changes the IPv6 source address to its own "outside" IPv4 address. The resulting IPv4 packet is then forwarded to the destination.

When the IPv4 destination responds, the IPv4 packets pass again through the NAT-PT which does the translation in reverse, replacing the IPv4 header with an IPv6 header and referencing its mapping table to find the correct IPv6 source and destination addresses. The upshot of this process is that the IPv6 host on one side of the NAT-PT thinks it's talking to another IPv6 host, and the IPv4 host on the other side thinks it's talking to another IPv4 host. Neither requires any specialized application or configuration.

While the mechanism sounds straightforward, there is a fair amount of complexity behind it due to the fact that IPv4 headers and IPv6 headers do not have a one-to-one correspondence of their component fields, such as IPv4 checksums. Adjustments between field values must be made in some cases, and the use of IPv6 extension headers can further complicate header translation. The data within ICMP messages, which can differ extensively between IPv4 and IPv6, must also be translated.


SA: 192.168.24.113
DA: 192.16.5.4
DATA
4host
172.16.5.4
SA: 2001:db8:25:1::345:29c2:15:abcd
DA: 2001:db8:192:15:bc16:270:1:20
DATA
SA: 172.16.5.4
DA: 192.168.24.113
DATA
SA: 2001:db8:192:15:bc:16:270:1:20
DA: 2001:db8:25:1::345:29c2:15:abcd
DATA
6host
2001:db8:25:1:345:29c2:15:abcd

A DNS application Level Gateway (ALG) in a NAT-PT translates

NAT-PT also presents a number of logistical challenges:

- For translation to operate correctly, DNS messages must pass through the translator. Querying DNS across an alternate path – or even through a different translator – will cause the translation to fail because NAT-PT will not have the information required to complete the translation.
- Careful consideration must be given to DNS server placement and to the addresses DNS is providing.
- Packets passing one direction through the translator must pass in the opposite direction through the same translator. Therefore network traffic patterns must be carefully controlled to prevent asymmetric patterns, potentially limiting design choices.
- The requirements of traffic to be bidirectional through the same NAT-PT means that the device on which it runs (usually a router or firewall) represents a single point of failure and an inviting attack target, reducing network reliability.
- The same issues around the disruption of applications that reference IP addresses in the upper layer by traditional IPv4/IPv4 NAT also apply to NAT-PT.

- NAT-PT cannot translate IP multicast traffic.
- Fragmented packets normally cannot pass through NAT-PT due to the differences in the way IPv4 and IPv6 handle fragmentation.
- The stateful nature of NAT-PT can cause timeout problems for some applications. For example, an application that might have long “silent” periods under normal operation might be required to implement some sort of keepalive mechanisms to insure that NAT-PT does not time out its address binding.
- NAT-PT implementations have shown scaling limitations in practical deployments.
- Capabilities that are supported by IPv6 but not IPv4 such as flow labels and Mobile IPv6 might cause confusion and failure if an IPv6 device attempts to use one of these features with an IPv4 device.

For these reasons, NAT-PT should be considered a specialized solution to be applied only when there are no other practical alternatives. For example, most IPv6-capable devices are also dual stack capable; such devices, as long as IPv4 addresses of some sort are available, can talk to IPv4-only devices, eliminating any need for translation. For applications that should be IPv6 accessible, proxies have proven better solutions due to their ability to address application-specific translation problems.

In fact NAT-PT has been challenging enough that the IETF has now deprecated it to historical status and recommends finding alternate solutions. Nevertheless vendors that have developed NAT-PT are unlikely to soon deprecate it in their own products, leaving the mechanism as an available solution when necessary.

Recent Developments in Implementation Mechanisms

The implementation mechanisms discussed in this paper are by no means the only tools that will ever be available. New mechanisms continue to be proposed; some will not gain enough interest to be developed, some will be developed but proven impractical, and some will become useful additions to the IPv6 implementation toolbox.

Among the newer proposals currently being discussed are:

- A stateless address mapping mechanism called IVI, which uses IPv4 addresses embedded in IPv6 addresses as a scalable alternative to NAT-PT. IVI has been used in CERNET2, the CNGI-sponsored Chinese academic and research network, for over two years.
- The IETF Softwires Working Group is presently proposing new solutions for interconnecting IPv4 and IPv6 networks, with a focus on tunneling IPv4 over IPv6.

- Comcast has proposed a mechanism called Dual-Stack Lite that addresses both the growing scarcity of IPv4 addresses and the need for existing IPv4 devices to communicate with new IPv6 devices. Rather than have traditional NAT devices at each site, IPv4 would be tunneled to more centralized carrier-grade NATs which both assign IPv4 to dual stacked devices and decapsulate IPv4 packets from IPv6 packets.

The success of these and other proposals is yet to be seen, but they serve as an assurance that means for overcoming the currently recognized challenges of IPv6 implementation continue to be developed.

Conclusion

The rapid depletion of IPv4 addresses means IPv6 is coming, whether we like it or not. Organizations worldwide resisted implementing IPv6 as early as they should have, because no business case could be made for it. What those same organizations are now realizing is that the business case for IPv6 is the ability to stay in business. IPv6 is an infrastructure issue, not a means of creating new revenue streams. And as activity around the world shows, the businesses and agencies that are the most dependent on the ready availability of globally routable IP addresses are actively working on deploying IPv6 in their networks.

Implementing IPv6 can be challenging under any circumstances. But with the right planning and the right choices of methodology and implementation tools, the costs and risks associated with an implementation project can be controlled. Juniper Networks, with its long history of high-performance high-reliability IPv6 support, remains the preferred choice for the most demanding next-generation networks around the world.

For Additional Information

For additional information, please visit the IPv6 Information Hub:
http://www.juniperipv6.com/

For additional information about JUNOS Software, please visit:
http://www.juniper.net/junos.

About Juniper Networks

Juniper Networks, Inc. is the leader in high-performance networking. Juniper offers a high-performance network infrastructure that creates a responsive and trusted environment for accelerating the deployment of services and applications over a single network. This fuels high-performance businesses. Additional information can be found at www.juniper.net.

CORPORATE AND SALES HEADQUARTERS
Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, CA 94089 USA
Phone: 888-JUNIPER
(888-586-4737)
or 408.745.2000
Fax: 408.745.2100
www.juniper.net

APAC HEADQUARTERS
Juniper Networks (Hong Kong)
26/F, Cityplaza One
1111 King's Road
Taikoo Shing, Hong Kong
Phone: 852.2332.3636
Fax: 852.2574.7803

EMEA HEADQUARTERS
Juniper Networks Ireland
Airside Business Park
Swords, County Dublin, Ireland
Phone: 35.31.8903.600
Fax: 35.31.8903.601

710085-001 Jan 09