SPIRENT Communications

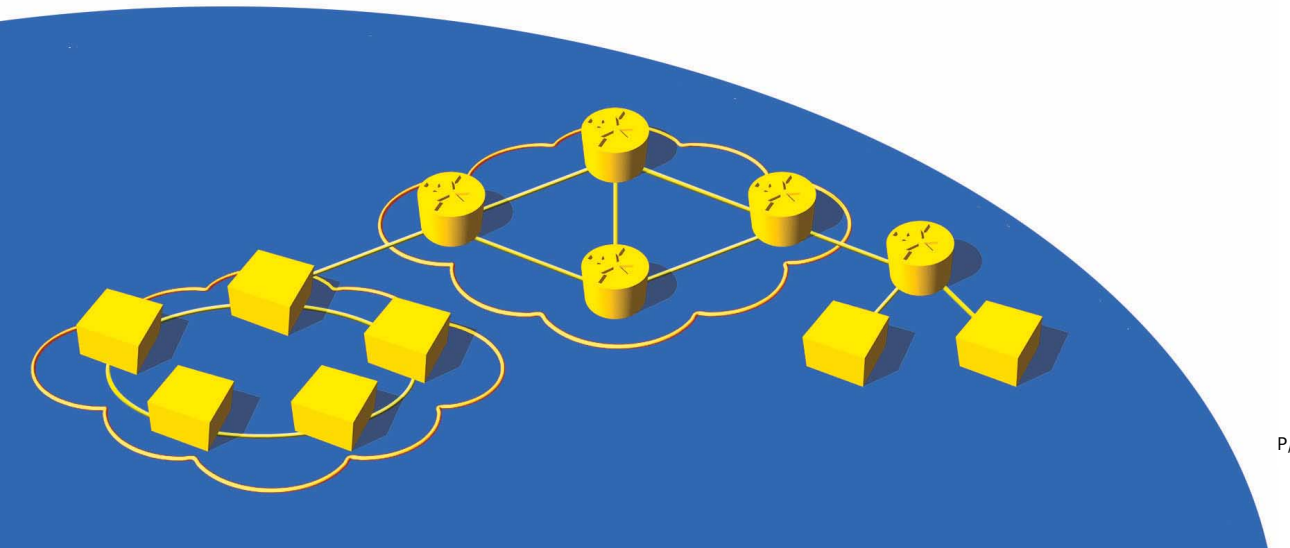
Analyze | Assure | Accelerate™

White Paper

IPv6 and the Next Generation Internet

Protocol Overview

July 2004



P/N 340-1293-001 REV A

Spirent Communications, Inc.

26750 Agoura Road
Calabasas, CA
91302 USA

Email: productinfo@spirentcom.com
Web: www.spirentcom.com

North America

+1-800-927-2660

Europe, Middle East, Africa

+33-1-6137-2250

Asia Pacific

+852-2166-8382

All Other Regions

+1-818-676-2683

IPv6 and the Next Generation Internet—Protocol Overview

This white paper discusses IPv6 testing and the future of the Internet and provides an overview of the protocol.

Contents

Introduction 2

IPv6 Overview 2

Transitional Protocols 6

Routing 8

Other IPv6 Developments 10

Implementation Challenges 11

IPv6 Testing 12

Conclusion 14

References 15

Introduction

The Internet is huge; it is ubiquitous. It has become a truly massive entity. The Internet is also in constant motion, growing and expanding at an unpredictable rate and in numerous technical directions. This growth is fueled by the world's insatiable demand for instantaneous information, entertainment, and communication. Left unchecked, this growth will continue to transform the once manageable Internet into an endless labyrinth of networks and computers. Yet, unchecked it remains. In fact, the advent of cellular Internet services will greatly exacerbate this problem. The potential infusion of half-a-billion new mobile Internet devices will finally break the current paradigm.

The world has nearly outgrown the current Internet, and cellular Internet devices will further expedite this process. A whole new Internet is needed. This new Internet must be built from the ground up. Like all new construction projects, the next generation of the Internet must be built upon a solid foundation. That foundation consists of a completely rewritten Internet Protocol (IP) known as IPv6.

IPv6 is designed, first and foremost, for scalability. We will never run out of IPv6 addresses – every person on the planet can have trillions of individual addresses. IPv6 also adds many other inherently useful communications features such as security, automatic configuration, and expedited classes of traffic. Together, all of these features set the stage for a new and improved Internet.

The new Internet model based upon IPv6 is revolutionary. However, changing to IPv6 must be an evolutionary process. The "old" Internet cannot be cast aside for the new – instead, both will need to coexist and interoperate for an extended transitional period. Furthermore, IPv6 is still entirely untested and unproven. The first IPv6 users (of which there already are a few) are truly pioneers. This new networking architecture must be validated prior to any large-scale deployments. Finally, IPv6 by itself is not the entire solution to the Internet's growth pains. Instead, IPv6 will be required to support many higher-level protocols and applications.

The Internet is in trouble, but IPv6 is a remedy. This remedy is neither immediate nor foolproof, so users will be well advised to proceed with caution. This paper provides a detailed analysis of the IPv6 protocol. Included in this analysis are ways to validate the functionality, performance, scalability, and interoperability of this up-and-coming new protocol.

IPv6 Overview

This section presents an overview of the IPv6 protocol and functionality. All of the basic IPv6 features including routing and transitional mechanisms are covered. This does not include detailed information about packet formats or exhaustive protocol interaction charts. For these additional details, please refer to the RFCs listed on *page 15*.

Addresses

The current version of the Internet Protocol (IP) is IPv4. IPv4 uses 32-bit binary addresses that are commonly converted to the familiar "dotted decimal" notation. The decimal representation of an IP address typically appears as 168.100.1.1. Since there are 32 bits in the address, simple binary arithmetic indicates that there are 2^{32} (approximately 4 billion) available addresses.

IPv4 was developed over 20 years ago. At that time, PCs were in their infancy, the planet's population was about 4.5 billion people, and it was inconceivable that the world would ever require more than 4 billion IP addresses. Unfortunately, those addresses were not allocated equitably. A large percentage of the IPv4 addresses are designated for special purposes. Furthermore, the early corporate and educational pioneers of the Internet such as MIT, Xerox, and Apple each have more available addresses than the entire country of China.

IPv6 addresses are 128 bits long. In scientific notation, 2^{128} is approximately 3.4×10^{38} addresses, or 7×10^{28} addresses for each person on earth. In short, even with any allocation anomalies that may occur in the future, there is no way that all of the IPv6 addresses will ever be depleted. IPv6 addresses are represented in hexadecimal notation with colons as delimiters, so a typical address looks like this:

21DA:00D3:0000:2F3B:00AA:00FF:FE28:9C5A

There are some shortcuts available that can compress parts of this address, but in general, IPv6 addresses are quite long and unwieldy.

The length of IPv6 addresses affords a high degree of flexibility when it comes to allocating and segmenting addresses. Several different proposals exist for dividing IPv6 addresses between network identifiers and host addresses. (IPv4 uses subnet masks to accomplish this function.) In most cases, at least the first 64 bits are reserved for network addresses, with the remaining bits containing the host address. In some cases, a device's existing IPv4 address or MAC (Ethernet) address can be embedded in the space designated for host identifiers. Each 128-bit address will be accompanied by a prefix length metric. This is directly analogous to an IPv4 subnet mask. The prefix length indicates how many bits (starting from the left-most portion of the address) will be used for the network portion of the address, while the remaining bits will identify the individual host computer's address. In order to build large and scalable networks, the IPv6 authors envisioned a hierarchical distribution of the network portion of the address to facilitate high levels of aggregation.

The first few bits of an IPv6 address indicate the packet type. Unicast, multicast, broadcast, and anycast (a new type) packets are fully supported by IPv6. The next portion of the address will then identify the network affiliation of that particular address.

IPv6 Packet Headers

The header of an IPv4 datagram contains several important pieces of data. The source and destination addresses are included in the header. Other fields in the header include the header length, type of service field, protocol identifier (for layer 4 functions), a flag field, a time-to-live counter, a checksum field, and then a variable quantity of optional facilities. All of these fields can be used by routers or end-stations for processing the data packet.

Although all of the fields in an IPv4 packet have specific purposes, not all of the fields are necessary in each packet. The IPv6 protocol streamlines the packet header by eliminating any unnecessary information. The first step is the elimination of the error detection facilities – the checksum and the flags. Upper layer protocols (layer 4 and above) have their own error detection and correction mechanisms. Therefore, another checksum at the IP layer is redundant. Thus, these fields have been eliminated altogether.

Two other simple changes in the IPv6 header are: First, the IP version number for IPv6 packets will be the binary representation of number 6 instead of 4. This information is contained in the "version" field, which is the first part of the IP packet header. The second change in the header is that the "time-to-live" field has been renamed "hop limit."

The 8-bit IPv4 type of service (TOS) field has been renamed for IPv6's purpose and is now called the "traffic class" field. It is expected that traffic classes will closely mirror the IPv4 usage of the TOS bits for differentiated services. A new 20-bit "flow label" field has been added to the packet header to facilitate the identification of a specific traffic stream. This flow can be associated with a particular traffic class.

The IPv4 protocol field has been eliminated. In its place, a new field called the "next header" identifier has been created. The next header is used to identify any options that may be included in the IP header such as fragmentation, routing, or security information. This optional data is only included if necessary. The next header field can also provide information about the layer 4 protocol such as TCP or UDP.

The "header length" field has been eliminated since all IPv6 headers are 40-bytes long. This parameter has been replaced with a "payload length" field, which indicates the entire length of the datagram. IPv6 supports datagrams up to 64 kilobytes in length. The size of the datagram also includes any IPv6 extension headers.

The final and most obvious change between IPv4 and IPv6 packet headers is the size of the addresses. The source and destination addresses have both been extended from 32 bits to 128 bits (16 octets) in order to accommodate the new expanded length of IPv6 addresses.

Extension Headers

Supplemental information can be added to IPv6 packet headers by using extension headers. Several extension headers can be strung together to support multiple options. Each extension header includes a "next header" field. This field is the same parameter that is included in the basic IPv6 header, and it points to the next extension header or to the layer 4 (typically UDP or TCP) header. All extension headers follow the IPv6 header

(directly after the destination address) and precede the data portion of the packet. The order of multiple extension headers is specified in RFC 2460. (For a complete list of the RFCs referenced in this document, see *page 15*.) Several extension headers have been defined by the IETF. Others are likely to be created as new requirements develop. The current extension headers (listed in order) are as follows:

- Hop-by-hop options – These options must be read by every router from the source to the destination. So far, only two hop-by-hop options have been defined. The first type is called "jumbograms" which are packets that exceed the maximum IPv6 packet size of 64K. These jumbo packets can be up to 4 Gigabytes in length. The second type of hop-by-hop options are known as "router alerts" – these are used to indicate that the packet is destined for the router itself, instead of for an endstation on one of the attached networks.
- Destination options – These are options that will be processed by the destination router only. No specific destination options have been defined at this time.
- Routing header – This replaces the IPv4 source routing option. This header provides an ordered list of all of the routers that must be traversed from source to destination. This can be used for traffic engineering implementations (such as RSVP-TE).
- Fragment header – The fragment header is added by the source computer. This header (like the IPv4 fragmentation option) contains the fragment offset, more fragments, and ID fields. The fragmented packet can then be reassembled by the recipient.
- Authentication header and encapsulating security payload header – These extension headers support payload security functions. These can work hand-in-hand using keys and sequence numbers to certify the integrity of the data. IPSec is included in the IPv6 protocol, and the associated keys will be contained in the encapsulating security payload header.

Some, all, or none of these extension headers can be implemented, depending upon the particular transport requirements associated with a given datagram. The last extension header's "next header" field will point toward the upper layer protocol header. For example, a "next header" value of 6 indicates that a TCP header will follow.

Neighbor Discovery Protocol

IPv6 is defined by the Internet Engineering Task Force (IETF) in RFC 2460. However, that is just the tip of a huge networking iceberg. Over 50 other RFCs define the features, capabilities and methodologies associated with IPv6 (see *page 15* for a list of related RFCs). One of the most important ancillary features is the neighbor discovery process, specified in RFC 2461.

The neighbor discovery protocol (NDP) is designed to supplant the IPv4 address resolution protocol (ARP). In addition to discovering the link-layer addresses of neighbors (which is the sole purpose of ARP), NDP also provides some other services. NDP can be used to discover the local routers, instead of requiring workstation users to explicitly configure default gateway addresses. NDP can also be used to learn the network prefixes of devices.

NDP uses a revised version of the Internet Control Message Protocol (ICMP) for its protocol foundation. Neighbor solicitation and neighbor advertisement messages are used to detect and maintain neighbor relationships. Similar router discovery advertisements are also used.

Autoconfiguration

Some extensions to IPv4 were developed (primarily by Microsoft) to facilitate the automatic assignment of IP addresses. This is accomplished using the Dynamic Host Configuration Protocol (DHCP). While there are IPv6 extensions available for DHCP, the preferred method is to use a new autoconfiguration process.

Similar to DHCP, the IPv6 autoconfiguration procedure is used by workstations. Instead of using a server for address distribution, the IPv6 autoconfiguration process is supported by a host's local router. The router will supply the workstation with an IP address, prefix, and default router information. Both "stateful" and "stateless" autoconfiguration methods are available, depending upon the level of control that the network administrator desires over the distributed addresses.

Transitional Protocols

The entire world cannot possibly convert to IPv6 overnight. Instead, IPv6 is being implemented in phases, starting with small pockets of experimental networks. The overall transition of the Internet will take many years, perhaps a whole decade. During this prolonged period of time, both protocols will need to peacefully coexist and interoperate. Furthermore, all network resources must be accessible to users regardless of their version of IP. Several different mechanisms have been developed to support a transitional environment.

Various perspectives exist with regard to transitional methodologies. Encapsulation is one popular option. However, this is not necessarily a straightforward solution. Should IPv6 data be encapsulated in IPv4 packets so that they can traverse the current Internet? Or should IPv4 packets be encapsulated in IPv6 so that they will be well suited for the future incarnation of the Internet? Or should both protocols be encapsulated in a common third format to ensure mutual compatibility? Other alternatives also include translation between the two protocols or support for dual stacks on workstations, hosts, and routers.

Encapsulation Methods

Protocol encapsulation or tunneling is a simple concept, but it is dependent upon two underlying assumptions. First, two endpoints must be running a common version of IP (IPv6 in *Figure 1 on page 7*). Second, the ingress and egress points of the intermediate network (IPv4 in this case) must be able to encapsulate, route, and de-encapsulate the packets – this additional processing overhead reduces overall network performance.

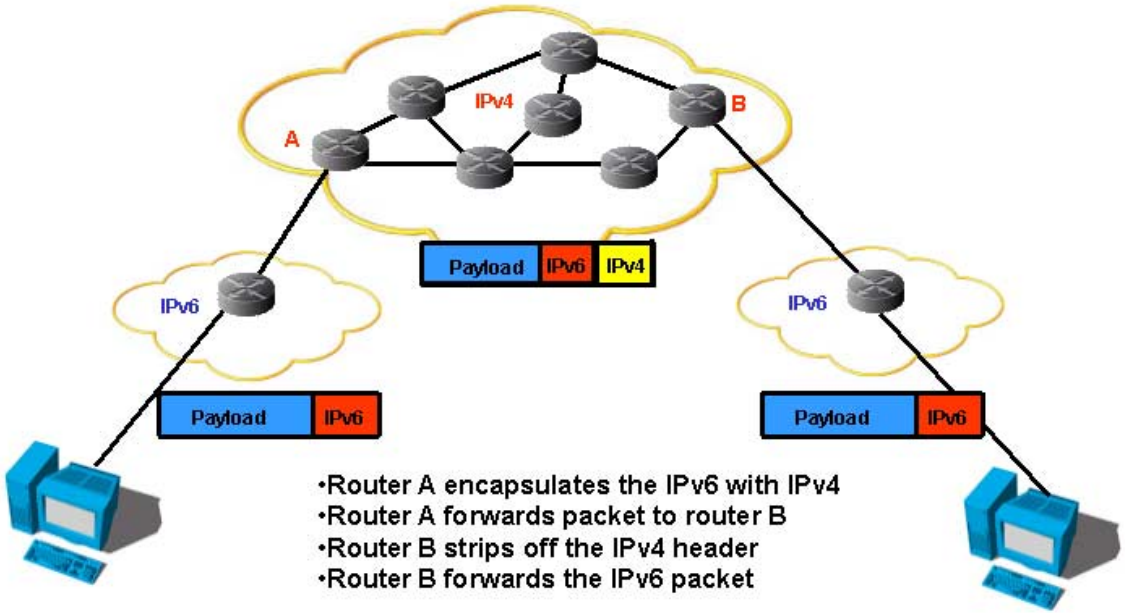


Figure 1. Network Tunneling Scenario

The diagram in *Figure 1* indicates a typical network tunneling or encapsulation scenario. In this case, two remote "islands" of IPv6 must be interconnected over the legacy IPv4 Internet. The source device on the left generates an IPv6 packet, which is destined for the right-hand workstation. At the entrance to the Internet (router "A"), an IPv4 header is added to the packet so that it can be routed using the traditional protocols. At the egress of the Internet (router "B"), the IPv4 header is removed, and a native IPv6 packet is delivered to the destination. Note that the end-stations are unaware of the intermediate protocol encapsulation; they simply operate as though they are using a direct IPv6 communications path.

Similar scenarios are also possible for connecting IPv4 "islands" over an IPv6 backbone.

Most tunnels are typically created manually as part of the basic router configuration for devices at the edge of the networks. However, manual configuration is not a very scalable mechanism, so automatic tunnels will be necessary in the future. A standard known as "6to4" specifies an automatic encapsulation and tunneling method. This specification includes a technique for mapping IPv4 addresses to IPv6 addresses.

Tunnels can be automatically propagated from a centralized tunnel broker, according to one proposal (RFC 3053). Another option that can be used in conjunction with 6to4 encapsulation is the Intra-Site Automatic Tunnel Addressing Protocol (ISATAP). This solution takes advantage of the autoconfiguration facilities inherent in IPv6 to distribute the appropriate tunnel addresses.

Dual Stack

Currently, the most common transitional mechanism is known as a dual stack implementation. This means that a workstation, server, and/or router must support both versions of IP. Therefore, it will be able to communicate with IP networks regardless of the version. Unfortunately, the overhead associated with dual stack implementations is quite substantial. Hosts or routers will need twice the processing power and memory resources that a single protocol would require. These extra resource requirements can seriously tax the performance and scalability of the device.

Translation

The third type of transitional methodology is simple network translation. This involves placing a gateway (typically a software application on a local router) between an IPv4 network and an IPv6 network. This gateway will convert packets from one version of IP to the other. This is the most effective method for true any-to-any communications; it ensures that all network resources are fully accessible regardless of the version of IP being used. Protocol translation should also be compatible with all of the other transitional architectures described in this document. The most common type of translation is known as Network Address Translation – Protocol Translation (NAT-PT).

Routing

The Internet Protocol (IP) operates at layer 3, the network layer, of the well-known OSI communications model. By definition, the network layer requires routing. Network reachability is determined by one or more of several different routing protocols. These protocols are mature and battle hardened. Most are in their third or fourth revision, so they have had the benefit of many years of use, improvements, and optimization in production networks.

New routing protocols have not yet been developed for IPv6. Instead, all of the major IP routing protocols (BGP, OSPF, RIP and ISIS) have modifications and extensions that allow them to support IPv6 networks and addresses. Over the years, these protocols have all been refined and optimized for IPv4 environments. IPv6 was an after thought. These protocols all have extremely limited practical experience in IPv6 environments (in fact, most router vendors are only just completing their OSPF IPv6 solutions). Furthermore, it is very likely that these protocols will also need to support transitional networks containing both IPv4 and IPv6. The performance, scalability, and functionality of legacy routing protocols in these new environments are not known.

BGP

The Border Gateway Protocol (BGP) is the routing protocol of the Internet. BGP is now in its fourth generation, so it is commonly referred to as BGP4 (the “4” does not reflect the version of IP that it supports). Since BGP was designed specifically for the Internet, it was built with scalability in mind. The current IPv4 Internet contains approximately 150,000 routes, and BGP handles this size network very effectively.

In 1999, some new fields were added to BGP update packets to allow the protocol to support any atypical Internet communications requirements that might be developed. In particular, these new facilities, known as the multiprotocol extensions, were initially supposed to be used to support multicast routing. These new fields can also be used to propagate IPv6 addresses. (Also, more recently, MPLS labels have been advertised using these BGP extensions.) The current IPv6 routes throughout the Internet are advertised via BGP. This revised version of BGP is sometimes called BGP4+ or MultiProtocol BGP.

OSPF

The Open Shortest Path First (OSPF) routing protocol is an interior gateway protocol, so it is typically used within a single autonomous system or administrative domain. As such, it is commonly found in large enterprise, academic, or governmental organizations. OSPF version 2 is the prevalent version, and it was designed specifically for IPv4 routes.

A new version of OSPF known as OSPFv3 was designed to support IPv6. OSPFv3 is very new in the industry and is not yet used in any major production networks. In general, OSPF is an extremely complex protocol – there are different functions and descriptions for many types of routers, network links, and advertisements. Many of these have been revised in OSPFv3; in fact, two entirely new link state advertisement types have been added. In general, it took approximately two years for most router vendors to fully debug their current OSPFv2 implementations. Although OSPFv3 is similar to its predecessor, the differences are substantial enough to cause one to assume that a lengthy shakeout period will be necessary for this complex reworked protocol. We are only at the very beginning of this debugging period.

RIP and ISIS

The Routing Information Protocol (RIP) has several scalability and functionality limitations. Accordingly, it is not commonly used in large networks. However, even smaller RIP-based networks will eventually need to migrate to IPv6. Therefore RIPng (“next generation”) has been developed to advertise IPv6 addresses. RIPng was developed in 1997; it was the first routing protocol to be modified for IPv6 support. RIP is a very simple protocol, and the IPv6 extensions are equally straightforward. RIPng, like OSPFv3 and BGP4+, is implemented only in very limited and exceptional circumstances.

ISIS is slowly gaining popularity in service provider networks. It too has been modified to support IPv6 – the new version is called ISISv6. Again, the rate of deployment for ISISv6 is quite minimal.

Overall, there are several different routing solutions for IPv6. However, real world experience with any of these IPv6 routing protocols is extremely minimal. The performance and scalability aspects are unknown. Furthermore, it is likely that these are only temporary band-aid style solutions. Longer term, a more practical solution will be the development of an entirely new routing protocol optimized specifically for IPv6.

Other IPv6 Developments

IPv6 promises many new features, all of which will incrementally improve the art of networking. IPv6 is poised to deliver all of its promises. IPv6 and its ancillary protocols have the potential to radically transform networking as we know it. Other benefits provided by IPv6, including data security and quality of service, are discussed in the next sections.

Security

The world's current geopolitical climate certainly punctuates the need for improved data security. Additionally, legislation, internal corporate requirements, and just plain common sense also dictate stringent information security requirements. However, the Internet is the antithesis of a secure network. It is an entirely open public network. The architects of IPv6 have chosen to directly address this issue of Internet security.

IP Security, commonly abbreviated as IPSec, is accomplished via encryption. The IPv6 encryption algorithms are identical to those used for IPv4. There are a couple of different options for encryption types and levels, but in general, an end-to-end secure tunnel is constructed through the Internet. IPSec can be added, as needed, to some IPv4 routers, servers, and workstations. Alternatively, a separate external tunnel server can also be used to initiate or terminate IPSec tunnels.

Instead of viewing IPSec as an adjunct function, IPv6 has built-in support for this secure protocol. A separate security extension header facilitates encryption and the associated key authentication. However, it is important to note that there is a significant trade-off that must take place in order to implement IPSec. The encryption and decryption functions (especially when 128-bit keys are used) require a lot of processing overhead and can seriously impact the performance and scalability of the associated devices.

Quality of Service

IPv6 provides a much more sophisticated quality of service mechanism than that of its predecessor. For starters, the IPv6 header contains an 8-bit traffic class field. This field can be mapped directly to the IPv4 type of service field. In short, this supports the same types of differentiated services as an IPv4 packet.

IPv6 augments the traffic class field with a 20-bit flow label field. This is an entirely new concept. Though the use of this field is not yet fully defined, the basic principle is quite intriguing. This label can be used to identify an end-to-end traffic flow, which would then, in theory, be assigned to a specific class of service. This could facilitate a deterministic end-to-end connection-oriented service through the connectionless Internet. This technique could substantially improve the quality of real-time audio and video applications.

The architects of IPv6 have gone to great lengths to ensure that it is a fully extensible and open protocol. Other new features can be added as they are developed in the future. The list of services that are supported by IPv6 can be expected to grow significantly over the coming years.

Implementation Challenges

IPv6 is the future of the Internet. It is not an option – it is an absolute necessity. However, it is not going to be a simple or rapid transition. IPv6 represents cyberspace's new and unexplored frontier. Many challenges, some of which are recognized and others that are still unknown, will accompany the implementation of this new protocol. Router manufacturers and service providers are cautiously "testing the waters" with new IPv6 implementations.

There are many unanswered questions associated with IPv6. Router vendors have had two decades to optimize, improve, and debug their IPv4 implementations. IPv6 will certainly benefit from these experiences, but it still has a long way to go before it is the feature rich, stable, and highly scalable protocol that its architects envisioned. Today's IPv6 Internet only contains approximately 1,000 routes (compared to 150,000 IPv4 routes) so real-world scalability and performance metrics have not been developed.

Service providers typically look to their equipment vendors for performance data. However, network equipment manufacturers' glossy data sheets and web sites tend to quote "best case" scenarios developed in controlled and sterile environments. Furthermore, the performance metrics that are quoted are almost always based upon IPv4. If IPv6 statistics are actually provided, it is still unlikely that they will take into account a transitional or mixed network.

An example of the substantial differences between IPv4 and IPv6 performance is illustrated in *Figure 2 on page 12*. This simple graph shows the latency associated with a dual stack IPv4/IPv6 router. The latency in microseconds is noted on the horizontal axis, and the quantity of received packets for a given latency value is indicated on the vertical axis. Note the consistency of the IPv4 measurements and the extreme variance associated with the IPv6 latency measurements. This deviation was noticed to further increase as IPv6 prefix lengths were expanded. Similar results were observed for several different routing platforms.

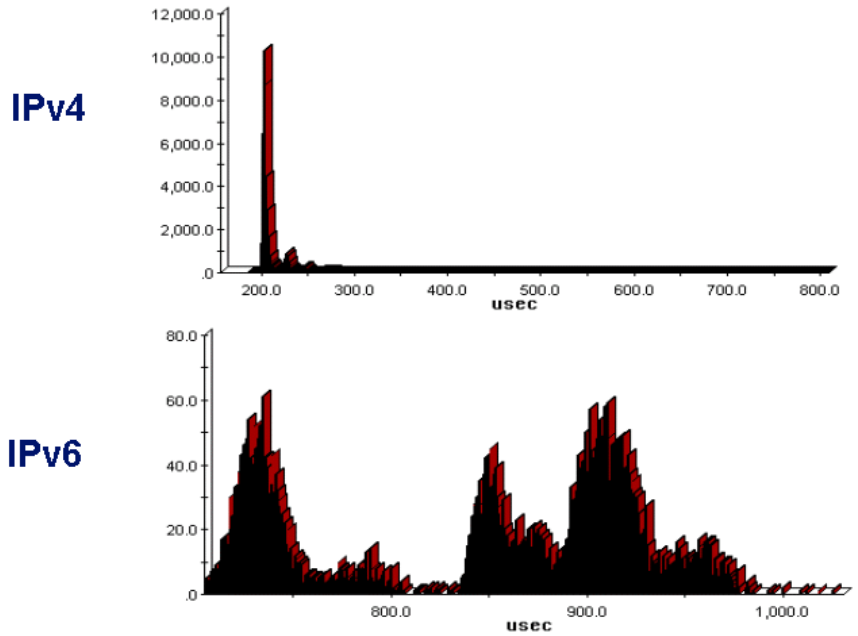


Figure 2. Latency with a Dual Stack IPv4/IPv6 Router

Graphs such as this one are quite troubling to network architects and real-time application designers. This data, from just a single router, indicates that it is impossible to predict the overall latency associated with an end-to-end IPv6 connection. Deterministic QoS calculations (or even estimates) will be entirely unattainable. Time-sensitive applications (for example, Real Audio or Quick Time Video) could be seriously impacted by the IPv6 network's inconsistent or outright poor performance.

Over the next several years, it is likely that router vendors will endeavor to greatly improve their IPv6 performance and functionality. In the mean time, service providers and enterprises are advised to thoroughly test IPv6 in their labs prior to deploying this new technology in their mission-critical networks.

IPv6 Testing

The old adage, "look before you leap" has never been more relevant to the networking industry than with the advent of IPv6. Many changes are coming, and most of them are good. However, all of these changes will impact the network users and applications in some way. Therefore, the same fundamental principles that users have lived by in past years still apply today – test, test, and test. IPv6 testing can be divided into four basic categories: conformance, functional, performance, and application. Each of these testing categories is explained in the following sections.

Conformance Testing

IPv6 is an evolving protocol. Over 50 new drafts on various aspects of IPv6 are currently being reviewed by the Internet Engineering Task Force (IETF). Services such as traffic flows, security, and other options are still under development. Many modifications to the existing RFCs are also likely to occur over the next few years. It is very challenging for equipment manufacturers to keep pace with these developments. Furthermore, the interpretations of the specifications of each new draft will vary from vendor to vendor. This, of course, leads to major interoperability challenges for end users. Conformance testing is the only way to ensure compliance with the accepted standards and drafts.

Conformance testing entails comparing a vendor's protocol implementation directly to the associated standard documentation. A neutral third party's interpretation of the specification is used as the common reference point. This interpretation is refined and tuned by incorporating input from a wide range of equipment manufacturers and industry experts.

The tester will need to select the appropriate RFCs for their particular needs. Certainly conformance with RFC 2460 (the actual IPv6 protocol) will be necessary. Conformance tests for additional functions such as autoconfiguration and neighbor discovery are also recommended. Routing protocols and transitional mechanisms should also be thoroughly tested if they are included as part of the desired networking solution.

Network equipment manufacturers must run a full battery of conformance tests for each new protocol and code revision that they develop. Service providers would be prudent to validate the conformance claims of each manufacturer (and each subsequent software release) prior to deploying any IPv6 devices in their network.

Functional Testing

The functional aspect of testing focuses on the operation of all of the concurrent features associated with IPv6. Among other things, transitional mechanisms represent a critical functional characteristic that must be carefully validated. Some recommended IPv6 functional tests include the following:

- IPv4/IPv6 concurrent data plane forwarding operations
- Neighbor discovery process
- Autoconfiguration
- Transitional tunneling protocols
- Network Address Translation (IPv4/IPv6)
- Concurrent multicast and unicast forwarding
- Anycast operations
- Security and IPSec functionality
- Quality of Service mechanisms

- Extension header support
- IPv6 routing protocol functionality (BGP4+, OSPFv3, RIPng, ISISv6)
- Multicast listener discovery process.

Performance Testing

Most people associate performance testing with raw packet throughput measurements. This certainly is a major component of performance testing, but there are also many other facets. A device's throughput, loss, and latency measurements are critical. For IPv6, these metrics should be derived under several different conditions. IPv6 prefix lengths should be varied to see how this affects the results. VLANs or ATM VCCs should be used to emulate real world networking scenarios, and the appropriate transitional mechanisms (such as the dual-stack chart shown in the previous section) should be invoked.

Performance testing is also used to ascertain a device's scalability. Maximum VLAN, VCC and VPN quantities should be determined through iterative testing processes. Routing tables (based upon the appropriate routing protocols) limits should also be identified by performance testing. Different combinations of extension headers should also be tested. As these control plane functions are modified, the data plane throughput, loss, and latency should be monitored continually. A fully automated or interactive test tool is highly recommended for all performance and scalability tests.

Application Testing

The final, and perhaps most important, aspect of IPv6 testing should take place at the application layer. Actually running (or simulating) the users' applications and data traffic over the IPv6 network is the only way to ensure that the users' experiences will be satisfactory. Applications should be mixed and matched (Email, work-flow applications, video conferences, etc.) based upon "worst case" scenarios. Control plane changes should be invoked while the applications are running in order to determine the applications' responses to real world network events.

Conclusion

IPv6 equipment manufacturers, industry forums, and seminars go to great lengths to describe all of the benefits that will be realized once IPv6 is implemented. All of these discussions seem to focus on the future. The overwhelming consensus throughout the networking industry is that IPv6 will be wonderful. The disagreement centers on the time frame and the implementation process.

As of yet, IPv6 is still new and unproven. Users should begin to develop IPv6 migration plans, but they should also move with caution. Thorough testing of the protocols, performance, and applications is critical. The ultimate success or failure of an IPv6 network is directly related to the level of testing that takes place prior to deployment.

References

The primary IPv6 specification is RFC 2460 – *Internet Protocol, Version 6 (IPv6) Specification* – December 1998

Other related RFCs are listed here:

- RFC 1719 – *A Direction for IPng* – December 1994
- RFC 1726 – *Technical Criteria for Choosing IP the Next Generation* – December 1994
- RFC 1752 – *The Recommendation for the IP Next Generation Protocol* – January 1995
- RFC 1809 – *Using the Flow Label Field in IPv6* – June 1995
- RFC 1881 – *IPv6 Address Allocation Management* – December 1995
- RFC 1887 – *An Architecture for IPv6 Unicast Address Allocation* – December 1995
- RFC 1888 – *OSI NSAPs and IPv6* – August 1996
- RFC 1981 – *Path MTU Discovery for IP version 6* – August 1996
- RFC 2080 – *RIPng for IPv6* – January 1997
- RFC 2185 – *Routing Aspects of the IPv6 Transition* – September 1997
- RFC 2292 – *Advanced Sockets API for IPv6* – February 1998
- RFC 2373 – *IP Version 6 Addressing Architecture* – July 1998
- RFC 2374 – *An IPv6 Aggregatable Global Unicast Address Format* – July 1998
- RFC 2375 – *IPv6 Multicast Address Assignments* – July 1998
- RFC 2450 – *Proposed TLA and NLA Number Assignments* – December 1998
- RFC 2452 – *IP Version 6 Management Information Base for the Transmission Control Protocol* – December 1998
- RFC 2454 – *IP Version 6 Management Information Base for the User Datagram Protocol* – December 1998
- RFC 2461 – *Neighbor Discovery for IP Version 6 (IPv6)* – December 1998
- RFC 2462 – *IPv6 Stateless Address Autoconfiguration* – December 1998
- RFC 2464 – *Transmission of IPv6 Packets over Ethernet Networks* – December 1998
- RFC 2465 – *Management Information Base for IP Version 6: Textual Conventions and General Group* – December 1998
- RFC 2467 – *Transmission of IPv6 Packets over FDDI Networks* – December 1998
- RFC 2470 – *Transmission of IPv6 Packets over Token Ring Networks* – December 1998

- RFC 2471 – *IPv6 Testing Address Allocation* – December 1998
- RFC 2472 – *IP Version 6 over PPP* – December 1998
- RFC 2473 – *Generic Packet Tunneling in IPv6 Specification* – December 1998
- RFC 2474 – *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Header* – December 1998
- RFC 2491 – *IPv6 over Non-Broadcast Multiple Access (NBMA) Networks* – January 1999
- RFC 2492 – *IPv6 over ATM Networks* – January 1999
- RFC 2526 – *Reserved IPv6 Subnet Anycast Addresses* – March 1999
- RFC 2529 – *Transmission of IPv6 over IPv4 Domains without Explicit Tunnels* – March 1999
- RFC 2545 – *Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing* – March 1999
- RFC 2553 – *Basic Socket Interface Extensions for IPv6* – March 1999
- RFC 2590 – *Transmission of IPv6 Packets over Frame Relay Networks Specification* – May 1999
- RFC 2675 – *IPv6 Jumbograms* – August 1999
- RFC 2710 – *Multicast Listener Discovery (MLD) for IPv6* – October 1999
- RFC 2711 – *IPv6 Router Alert Option* – October 1999
- RFC 2732 – *Format for Literal IPv6 Addresses in URLs* – December 1999
- RFC 2740 – *OSPF for IPv4* – December 1999
- RFC 2765 – *Stateless IP/ICMP Translation Algorithm (SIIT)* – February 2000
- RFC 2766 – *Network Address Translation – Protocol Translation (NAT-PT)* – February 2000
- RFC 2767 – *Dual Stack Hosts Using "Bump-in-the-Stack" Technique (BIS)* – February 2000
- RFC 2780 – *IANA Allocation Guidelines for Values in the Internet Protocol and Related Headers* – March 2000
- RFC 2858 – *Multiprotocol Extensions for BGP-4* – June 2000
- RFC 2874 – *DNS Extensions to Support IPv6 Address Aggregation and Renumbering* – July 2000
- RFC 2893 – *Transition Mechanisms for IPv6 Hosts and Routers* – August 2000
- RFC 2928 – *Initial IPv6 Sub-TLA ID Assignments* – September 2000

- RFC 3041 – *Privacy Extensions for Stateless Address Autoconfiguration for IPv6* – January 2001
- RFC 3053 – *IPv6 Tunnel Broker* – January 2001
- RFC 3056 – *An Anycast Prefix for 6to4 Relay Routers* – June 2001
- RFC 3111 – *Service Location Protocol Modifications for IPv6* – May 2001
- RFC 3122 – *Extensions to IPv6 Neighbor Discovery for Inverse Discovery Specification* – June 2001
- RFC 3142 – *An IPv6-to-IPv4 Transport Relay Translator* – June 2001
- RFC 3146 – *Transmission of IPv6 Packets over IEEE 1394 Networks* – October 2001
- RFC 3178 – *IPv6 Multihoming Support at Site Exit Routers* – October 2001
- draft-ietf-ngtrans-isatap – *Intra-Site Automatic Tunnel Addressing Protocol*
- draft-ietf-ngtrans-dstm – *Dual Stack Transition Mechanism (DSTM)*
- draft-ietf-isis-ipv6 – *Routing IPv6 with IS-IS*
- draft-vida-mld-v2 – *Multicast Listener Discovery (MLD) version 2*


SPIRENT
Communications