



***North American IPv6 Task Force (NAv6TF) Technology Report
“Analysis of IPv6 Features and Usability”***

Version 1.0

9/6/06

Primary Author/Editor: Wesley M. Eddy

Contributing Authors: William Ivancic, Joseph Ishac

weddy@grc.nasa.gov

1. EXECUTIVE SUMMARY	3
2. INTRODUCTION	3
3. COMPARISON AND CONTRAST BETWEEN IPV6 AND IPV4 FEATURE SETS.....	4
3.1. Addressing and Routing	5
3.2. Quality of Service	8
3.3. Security	9
3.4. Mobility	10
3.5. Multicast and Anycast	11
3.6. Flexibility and Growth	12
4. FACTUAL DATA IN RESPONSE TO IPV6 MYTHS	13
4.1. Application Support	13
4.2. Maturity Level	14
4.2.1. Protocol Documentation	15
4.2.2. Running Code.....	16
4.2.3. Real-World Deployment	16
4.2.4. Policy Directives	18
4.3. Header Overhead.....	19
5. CONCLUSIONS / SUMMARY.....	21
6. ACKNOWLEDGEMENTS	23
7. NAV6TF DISCLAIMER.....	23
8. ABOUT NAV6TF	24
9. ABOUT THE AUTHORS	24
10. REFERENCES.....	24

1. Executive Summary

It is well understood that IPv6 has been designed to replace IPv4. However, many network architects and engineers are still unsure of what all of the potential business cases for building IPv6 networks and services are. There also seem to be festering doubts regarding deployment and transition issues. We have done a feature-by-feature comparison and contrast of IPv6 versus IPv4 and found that IPv6 offers many unique opportunities for increasing a network architecture's efficiency and agility.

Significant facets of IPv6 in relation to IPv4 include:

- IPv6 enables addressing architectures that scale well in terms of the number of nodes and subnetworks, the size of subnetworks, and the degree of change within subnetworks; this includes typically-encountered cases where IPv4 becomes difficult to use robustly.
- Global routing tables in IPv6 are potentially much simpler than their IPv4 counterparts, and thus require lower memory and computational resources.
- In resource-constrained environments, IPv6 requires less processing than IPv4, which can result in reduced power demands and latencies, especially for routers.
- The flow-label in IPv6 is an enabler for per-flow Quality of Service with simpler algorithms and more efficient implementations that also permit the remainder of a packet to be encrypted; all of which are precluded in IPv4.
- Network and device security is boosted in IPv6 based on address manipulation techniques and secure neighbor discovery features that have no IPv4 counterparts.
- Routing for mobile nodes is more efficient in IPv6 than in IPv4. Smooth handover techniques for IPv6 also exist with no IPv4 equivalents.
- Current standards activities indicate that many future features may be developed for IPv6, but not necessarily for IPv4.

We have also examined some of the commonly held views on why not to deploy IPv6, and for the most part, found these beliefs to be based on verifiably inaccurate information. Specifically, we have found that application support, available literature, hardware and software compatibility, current user base, and header overhead are mostly invalid concerns.

2. Introduction

Currently, two versions of the Internet Protocol (IP) are in use on the Internet. In some sense, there is a competition going on between these protocols, as they are not directly

compatible, and network providers and users are being forced to determine whether to support one or both protocols for various network services. IP version 4 (IPv4) is the incumbent and currently has the most widespread usage for conventional Internet applications. IP version 6 (IPv6) is a large-scale re-design and re-engineering of IPv4, based on many lessons learned as the IPv4-based Internet grew and was used in unforeseen ways. Although it would seem obvious that IPv6 is a superior and valuable protocol to deploy, there is often considerable resistance to enabling IPv6 because decision-makers have difficulty in seeing a business case for IPv6, unsure of how it can be less costly, more efficient, more productive, etc than the IPv4 status quo. Also, some analysts have propagated significant amounts of misinformation about IPv6 over the last several years.

This report has two main purposes. The first is to highlight the significant differences between IPv4 and IPv6, leading to key points that can be used in business case analysis. Section 3 tackles this by dividing the feature sets amongst several categories including: addressing and routing functions, quality of service (QoS) capabilities, security properties, node and network mobility functions, support for additional communications modes beyond unicast (multicast and anycast), and flexibility or growth considerations.

The second purpose of this document is to dispel some of the conventionally accepted misconceptions about IPv6. We do this in Section 4, focusing on some broad issues that we have seen raised multiple times in meetings; namely, these are concerns regarding applications, IPv6's usability as a mature and available protocol, and IPv6's packet overhead on the wire.

3. Comparison and Contrast Between IPv6 and IPv4 Feature Sets

The IPv6 standard was the output of the IPng course of action to find a replacement for IPv4, given all of the lessons learned from using IPv4 and the evolution of the Internet. While many people have the notion that the increase in address space is the main advance in IPv6, the content of this section shows that address size is only the tip of the iceberg. In fact, the IPv4 address size was only one of around a dozen IPv4 aspects that were seen as necessary to change. The search tool on the RFC Editor's website¹ can be used to find many RFCs that document the history of the IPng process. Archived copies of expired Internet-Drafts with further details can also be found on the Internet. In April of 2006, a simple search for "IPng" yielded 41 RFC documents, most of which are Informational and contain inputs to the IPng from representatives of various industry segments, governments, or research labs.

¹ See <http://www.rfc-editor.org> to make use of this tool.

The remainder of this section is meant to educate readers regarding some of the differences between IPv6 and IPv4, especially those that might influence the business cases for either deploying IPv6 in existing IPv4 networks or using IPv6 as the basis for new network architectures (as is being analyzed in the niches of aeronautical networking and space exploration, for instance). We present bare facts here, with references to the relevant documents (generally RFCs) that contain more background and further information.

3.1. Addressing and Routing

The most obvious difference between IPv6 and IPv4 is that IPv6 addresses are 128 bits [1], whereas IPv4 addresses are only 32 bits [2]. This increase in the raw number of bits means that there is a factor of 2^{96} more addresses available in IPv6 than in IPv4. Due to the way that the address spaces are subnetted, scoped, and defined for multicast, private/experimental use, and other factors, the actual contrast is less direct than this simple factor.

Aside from a few blocks set aside for local-use, multicast, or other specific functions, the majority of the IPv4's 32-bit address space is designated for global unicast addresses [3]. In the IPv4 addressing architecture², IANA delegates Regional Internet Registries (RIRs) /8 address blocks (8-bit network identifiers, also historically called "class A" address blocks), which the RIRs can then divide into variable-length blocks for further assignment to ISPs or other registries [6, 7]. In this regime, the maximum address block that a site can ever be given is a /8, which leaves only 24 bits for subnetting and addressing within the organization. Historically, large or complex organizations have required multiple /8s. For instance, at least 7 /8s belong to the US Department of Defense. Considering there are only 256 such blocks, the IPv4 address space can be seen as severely limited in its ability to provide unique addresses to the elements of large organizations worldwide. To compound matters, even using multiple /8s is a poor solution, since there is no guarantee that the blocks will be numerically continuous, and if they are not, then both the local numbering scheme may be awkward, and multiple global routing table entries will be stored and propagated for the same site. In recent years, many IPv4 users have circumvented these issues by using Network Address Translators (NATs), although this practice is known to be fraught with problems of its own [8, 9].

² Here we refer to the Classless Inter-Domain Routing (CIDR), which is the IPv4 addressing scheme adopted in response to the failure of the older class-based addressing system to scale in step with the growth of the number of end sites [4, 5].

According to the IPv6 addressing architecture [10], the prefix of 001 identifies IPv6 global unicast addresses, so 1/8 of the address space or 2125 such addresses are available. To date, IANA has given IPv6 address blocks varying from /16 to /23 in size to the RIRs. The documented policy for the downstream assignment from RIRs to Local Internet Registries (LIRs) is that each LIR receive a minimum of a /32, and the minimum-sized address block that an LIR can then give to a site is a /48 block³. Since an IPv6 site can expect at a *minimum*, a /48, this allows for 16 bits of subnetting space and 64 bits for interface identifiers within a subnet (80 bits combined). Contrast this to an IPv4 site that can expect a *maximum* of a /8 block, leaving only 24 bits of space to be used for subnetting and host addressing combined. Since in reality, the vast majority of IPv4 sites do not get /8s, but rather /16s or /24s, there are more likely to be only 4 to 8 bits left for identifying hosts within a subnet, using global addresses.

The IPv6 addressing architecture utilizes scoped addresses, including scoped multicast addresses. Support for scoping in IPv6 is more fully defined and has some features that IPv4 has no analogues to. As a simple example, IPv6 has all-routers addresses in every subnet, which allow a node to find or communicate with routers without knowing their unicast addresses ahead of time, and without having the packets be processed by other end hosts. In IPv4, this is not possible without the assistance of other protocols.

Configuring and managing addresses in IPv6 and IPv4 can be accomplished using versions of the DHCP protocol. Additionally, IPv6 has mechanisms that allow a node to configure its own globally routable address, without the need for DHCP [11], and IPv4 has no counterpart to this functionality. DHCP is still of practical use in both protocols though, due to its ability to provide other configuration data, such as DNS server addresses.

Due to the fact that LIRs assign subnet addresses in IPv6, rather than simply end-node addresses (as often done in IPv4), DHCP supports prefix-delegation extensions for IPv6. Prefix-delegation allows DHCP to manage the assignment of subnet prefixes in an automated fashion, and allows IPv6 routers to be automatically configured [12]. IPv4 has no comparable feature.

In conjunction with the depletion of the IPv4 address pool, a second major driver in the design of IPv6 is that IPv4 inter-domain routing tables are very large. This is due to the frequent inability to aggregate addresses based on the way that IPv4 blocks have been assigned. The IPv6 addressing architecture and assignment policy is designed such that subnet addresses can potentially be aggregated very effectively. Essentially, the

³ For more details, see the IANA's web page on IPv6 allocation policies, <http://www.iana.org/ipaddress/ipv6-allocation-policy-26jun02>.

global routing table only needs to know how to reach a small number of large backbone networks, and the subnet addresses belonging to millions of end-sites can be aggregated hierarchically under the backbone provider network prefixes. This prevents routers from using large amounts of memory on their routing tables, thus allowing lookups to be faster, and network operators to spend less money on expensive router memory upgrades. A second benefit is increased speed in the computation of routing tables, which could speed convergence times. Recent developments indicate that Provider Independent addressing may become more prevalent in IPv6 assignments, and so these features may be somewhat reduced in practice.

In the effort to build faster router platforms, two well-known speed bumps in IPv4 were performing the checksum operations and fragmenting datagrams, as required. While relatively efficient means of computing the IPv4 checksum [13], and even implementing it in hardware [14], were developed, it was decided to improve speed by not including any internetwork-layer checksum at all in IPv6. The rationale behind this is that most link-layer protocols have at least their own checksum or CRC, and often their own retransmission protocols and error-correcting codes. Furthermore, reliable transport and application protocols also implement additional checksums. Since many of the link and higher-layer checksums in use are actually more powerful than the simple IPv4 checksum (a mere 16-bit one's complement sum across the header), it is of relatively little utility.

Typical IPv4 router designs are incapable of performing fragmentation operations in their optimized "fast-path", and instead have to resort to the "slow-path" for processing of packets that require fragmentation. This can represent a bottleneck that limits throughput and loads the central processor, which is also used for routing table maintenance and general device control. Since this is exploitable merely by users at any point in the network sending packets larger than a particular link's MTU, this could be seen as a potential weakness. In IPv6, routers never fragment packets; packets larger than an outgoing link's MTU are dropped. It is an IPv6 source node's responsibility to proactively fragment its own packets.

The lack of checksum and fragmentation responsibilities potentially allow IPv6 routers to perform slightly faster and with lower power requirements, but these differences are likely to be fairly minimal under typical uses cases, only becoming apparent in resource-constrained environments.

Another distinction between IPv6 and IPv4 is in the way that IP addresses within a subnet are resolved into link-layer addresses for transmission. IPv4 uses the ARP [15] mechanism for this, while IPv6 uses Neighbor Discover (ND) [16]. There are at least two key differences between ARP and ND. The first is that ARP operates directly on top of the link layer, while ND operates using ICMPv6, on top of IPv6, on top of the link layer.

Practically, this means that in the design of link layer protocols, distinct codes separating ND and IPv6 packets do not have to be defined, whereas IPv4 and ARP require separate code-points. This is of only marginal importance. The main difference between ARP and ND is that IPv6's ND is highly extensible through IP and ICMPv6 options. This extensibility has been used for a number of purposes, including security (authentication of network elements and resolution protocol messages), automatic prefix and interface identifier configuration, and advertisement of link MTU. IPv4's ARP has no such facilities and no means for extension beyond variable length address fields.

Altogether, in terms of addressing, routing, and forwarding features, IPv6 has identifiable advantages over IPv4 in every respect considered here. The main points of this section, and the following sections, are summarized for the reader's benefit in Section 5.

3.2. Quality of Service

The Differentiated Services QoS architecture utilizes the IPv4 Type of Service byte and the IPv6 Traffic Class byte in the same way and with the same rules for contained values [17]. In this respect, IPv4 and IPv6 headers have equivalent functionality for use with Differentiated Services, which typically is used for QoS between defined aggregates of flows. Flow aggregates are typically marked by network devices that determine each packet's designation through policy rules and some level of inspection. This is in contrast to per-flow QoS, where individual flows are accounted for rather than aggregates.

Regarding per-flow QoS, a significant capability that is part of the standard IPv6 header, and is not present in the IPv4 header is the ability to classify traffic into flows based on a flow label header field [18]. This can be used as a basic building block to efficiently support QoS policies and protocols. In IPv4, individual flows can only be classified by the relatively expensive process of examining (and possibly parsing) header fields. In most cases, this consists of matching the IPv4 addresses and the field that identifies the payload protocol, along with the port numbers in the transport protocol header. When tunneling protocols or encryption protocols are used between the outer IP header and the transport header, either per-flow QoS fails, or additional difficult processing is required. This is one form of deep packet inspection, and known to be problematic and fragile to the deployment of new services and protocols. Thus, per-flow QoS in IPv4 networks is rather difficult, but can be enabled with relatively low computational overhead and architectural impact when using IPv6.

3.3. Security

Both IPv6 and IPv4 can be used in conjunction with the IPsec suite of protocols [19]. In fact, the operation of the IPsec protocols is basically identical whether they are being used with IPv4 or IPv6. Since the Transport Layer Security (TLS) protocol [20] runs over top of the transport layer, and does not interface directly with IP, it is also mostly agnostic to the version of IP that is used. Additionally, the X.509 format for certificates (often used in IPsec and TLS) has encoding methods for both IPv4 and IPv6 addresses [21]. So, the two most prevalent security architectures in the Internet suite, IPsec and TLS, have no significant differences in use between IPv4 and IPv6 networks.

It is popularly touted that IPv6 has superior security properties to IPv4. In the majority of cases, the main reasoning used to justify this claim is that IPsec is a part of IPv6, since in early IPsec specifications [22], it was stated that all IPv6-capable hosts MUST implement the IPsec Authentication Header in a basic configuration (keyed-MD5 with 128-bit key [23]). In contrast, supporting any part of IPsec was optional for IPv4 hosts, as IPsec did not come into being until well after IPv4 was mature and widely deployed. In fact, the marriage between IPv6 and IPsec is not this clear. Current IPv6 node requirements mandate that IPv6 nodes MUST support both the Authentication Header and Encapsulating Security Payload portions of IPsec [24]. In contrast, the IPsec architecture description in RFC 4301 is less strict, saying that IPsec nodes MUST support ESP, and MAY support AH (i.e. AH support is not mandatory). No RFC that we are aware of requires IPv6 nodes to support automated keying and association management for IPsec through IKEv2. In any case, it seems that IPv6 nodes are at least expected to have ESP support. However, since IPv6's core functions do not rely on IPsec, and only support for manual keying is required (IKEv2 support is advised but optional), the argument that IPv6 is more secure than IPv4 based on the requirement to support IPsec is not well-founded. The reality of the situation is that IPv6-conformant implementations in practice are more likely to have support for IPsec, but it is still up to the users and network managers to configure them, and the exact same IPsec features are also readily available in IPv4 implementations, but not required by IETF fiat to be present.

Outside of IPsec, there are other features of IPv6 that are not found in IPv4, and can potentially give IPv6 better security properties. A couple of the features included under the Network Architecture Protection umbrella [25] that are relevant to security are end-system privacy and topology hiding. End-system privacy refers to the ability of an IPv6 end-system to generate and change its own IPv6 address through selection of the Interface Identifier portion of the address. A node can use this capability to change its address periodically to avoid perceived threats such as easily being able to correlate its entries in remote log files of web activity [26]. IPv4 has no equivalent capability, mainly

because the number of bits past its subnet mask is too limited to allow this kind of technique within. IPv4 nodes can dynamically change their public addresses using DHCP, but DHCP servers are rarely configured to permit this, and it requires a DHCP server, whereas the IPv6 solution is end-host based. Topology hiding is a related technique for protecting subnets that involves changing the IPv6 address prefix referring to a subnet, rather than the interface identifier. This prevents an adversary from using a known host's address to compute related addresses of other potential targets.

In addition, IPv6 has the optional secure neighbor discovery extensions, which allow hosts to authenticate the ND protocol messages [27], and define cryptographically-generated addresses to prove address ownership with reduced requirements for certificate management or other security infrastructure [28]. IPv4 has no comparable features, and its address space is too small for these capabilities to be portable to IPv4.

In summary, IPv6 does have superior security properties in comparison to IPv4, but these have little to do with IPsec, and both the IPsec and TLS functionalities are equivalent without regard to the underlying IP version⁴.

3.4. Mobility

Support for node mobility is not required in either IPv4 or IPv6, however, both protocols support network-layer mobility extensions [30, 31]. The means of supporting mobility and the features of each mobility protocol differ in specifics. Mobile IPv4 uses UDP for signaling, whereas Mobile IPv6 uses IPv6 extension headers. This allows for a cleaner implementation, since the code can be fully integrated with the IP-processing where it belongs, and no transport protocol port numbers need to be bound for special use.

Mobile IPv4 has two basic modes of operation, triangle routing and bi-directional tunneling, both of which incur some inefficiency in that the direct round-trip path between mobile and corresponding nodes is not traversed. Mobile IPv6 supports an optional route optimization mode that is more efficient than the alternatives available in Mobile IPv4. Route optimization avoids both the header overhead of tunneling and the latency involved in the persistent routing indirection that Mobile IPv4 depends on for reaching mobile nodes. In certain scenarios, the Mobile IPv6 route optimization feature might also be more economical to operate, since it greatly reduces the amount of traffic to and from the home network.

A method for reducing the latency involved in restoration of service as a Mobile IPv6 node moves from connectivity with one access router to another has been defined in

⁴ A much more detailed discussion of IPv6 security can be found in Kaeo, Green, Bound, and Pouffary's NAv6TF Technology Report [29].

Fast Handovers for Mobile IPv6 (FMIPv6) [32]. The IETF has also produced Hierarchical Mobile IPv6 (HMIPv6) with similar stated goals to FMIPv6, but differing in method [33]. Both of these optimizations have been demonstrated to be effective, and have no analogous MIPv4 extensions.

Mobile routers (and correspondingly, the mobile networks behind them) are supported in Mobile IPv4, but their operation is not particularly well specified (an attempt to remedy this is work in progress by the IETF). In contrast, Mobile IPv6 mobile routers, called NEMO routers, are specified very clearly in their own standards documents [34]. Many of the complications and difficulties that arise only in mobile router scenarios, but not with simple mobile nodes, are only being solved actively in the IETF in the context of NEMO/Mobile IPv6, and not being worked on in the context of Mobile IPv4 (e.g. the products of the MONAMI6 working group for multiple simultaneous care-of address registrations [35]).

In summary, based on cleaner design, support for route optimization, smooth handover techniques, and NEMO extensions, IPv6 has superior mobility features to IPv4, and current IETF work seems like it will exacerbate this situation.

3.5. Multicast and Anycast

IPv6 and IPv4 are both capable of supporting network-layer multicast communications. The major differences between IPv6 and IPv4 in terms of multicast lie mainly in the fact that multicast support is considered an “additional” or tertiary part of IPv4, whereas in IPv6 it is integral. IPv6’s addressing architecture defines certain commonly useful multicast addresses (e.g. all-routers), and describes the ability to scope multicast addresses (e.g. there is a link-local scope that can refer only to neighbors on a link, along with scoped groups for interface-local, admin-local, site-local, organization-local, and global use) [10]. Scoped multicast is used as a building block for the ND service for autoconfiguration in IPv6. Broadcast addresses as used in IPv4 are replaced with multicast addresses of the appropriate scope in IPv6 protocols.

The basic host to router multicast protocol in IPv4 is IGMPv3 [36]. In IPv6, this function is filled by MLDv2 [37], which is functionally equivalent to IGMPv3. The main difference between the two protocols is similar to one of the differences between ARP and ND, in that IGMPv3 messages are encapsulated directly in IPv4 datagrams, whereas MLDv2 messages are carried by ICMPv6 messages inside IPv6 packets. The difference is in the reuse of ICMPv6 as a general-purpose control messaging and signaling protocol, rather than defining a new protocol number with separate processing from the link layer.

In theory, the IPv4 and IPv6 multicast features might be seen as comparable, but in reality IPv6 has a large advantage in that it was designed from the start with multicast

communications as a consideration. As deployed, IPv4 routers on the public Internet are usually not configured to support much if any multicast traffic, whereas IPv6 routers must support some levels of multicast to perform their own basic functions. IPv6 multicast also does not rely on the ungainly tunnels that are used in IPv4 multicast to get around the common one-to-one operating system mapping between interfaces and IPv4 addresses. Since IPv6 specifically supports assigning several addresses to a single interface, multicast support is more straightforward.

While the early work on the concept of anycast involved IPv4 [38], and in the IPv4 Internet, anycast is actively being used in particular niches [39], anycast features are not formally part of the IPv4 standard. In contrast, anycast is supported directly in the IPv6 standard. Technically, most unicast routing protocols can support anycast without any changes, since routing messages advertising anycast groups have similar semantics as those advertising multi-homed sites. However, due to the differences in nature between unicast and anycast communications, changes at other layers of the protocol stack are required to properly use anycast. Anycast continues to be a research area with several challenging topics. For example, it is not clear how IPv6 anycast will be used on a global scale [40]. Given the degree of uncertainty in what the utility of anycast is, and how technical barriers for global use could be overcome, it does not seem to be possible to assess IPv6 versus IPv4 anycast features at this time.

3.6. Flexibility and Growth

Enterprise network designers have a strong desire for their networks to be able to grow with an organization's needs and be flexible enough to allow for rapid deployment of new applications and services. IPv6 currently seems much more capable than IPv4 in meeting these demands. For instance, the limited amount of space available for subnetting in IPv4 makes networks relatively inflexible. Renumbering in IPv4 is a difficult operation that the protocol was not designed for, whereas IPv6's design has a number of features that allow automatic renumbering to be smoothly and efficiently supported [41].

As noted in Section 4.2, there is currently only one IETF working group that seems to be chartered to provide an IPv4-centric solution, while there are many groups working on IPv6-specific solutions. This indicates that in the future, it is possible that a number of new network layer enhancements may only be available for IPv6 networks. This is justified by the fact that the limited number of bits in IPv4 addresses precludes techniques that require clever manipulations of interface identifier bits. It should be noted however, that the vast majority of IETF groups are pursuing solutions that work in conjunction with both IPv4 and IPv6.

In many IPv4 end-sites, the use of NAT is popular for a number of reasons. However, NAT is known to have many poor architectural properties [8, 9]. In IPv6, the common

NAT functionalities that network administrators are interested in can all be performed without any of the negative repercussions [25]. The ability to deploy new applications without any concern for application layer gateways in the NAT, or complex tunneling mechanisms [42, 43] alone is a large practical benefit of IPv6.

4. Factual Data In Response to IPv6 Myths

As with many new technologies, there is substantial hype and also substantial amounts of misinformation that abound in conversations and trade publications. In this section, we examine three classes of rumors that we have encountered regarding IPv6. These are formulated in response to such notions as:

- “Most applications do not support IPv6.” Or, “It is difficult to port applications that use IPv4 to allow IPv6 or dual-stack operation.”
- “IPv6 is still a research toy, and not supported by industry.” Or, “IPv6 is still a rapidly changing work-in-progress and can’t be used because its definition is too fluid.” Or, “Nobody is using IPv6 or letting it be transported on their networks.”
- “IPv6 headers are so big that they reduce link efficiency.”

When these kinds of positions are encountered, it is often difficult to ascertain their technical accuracy, due to the information needed to definitively verify or disprove them being either scattered amongst many sources or difficult to quantify. We present evidence in this section that all three classes of rumor are decidedly false and not factually based.

4.1. Application Support

A healthy amount of documentation and examples are available on writing new applications and porting old applications to be capable of operating in any compatible combination of local and remote IPv4, IPv6, or dual-stack support. The IETF has produced some informational guidelines on this topic [44]. By following these clear guidelines it should be possible for old IPv4-only applications to be quickly ported into hybrids that function with IPv4, IPv6, or dual-stack connectivity within the same code. The main elements required for this functionality are part of the “basic extensions” to the BSD socket interface that is implemented by most common operating systems [45]. Additional “advanced extensions” that pertain to some specific IPv6 features have also been described in an informational RFC [46]. Additional documentation pertaining to

specific operating systems and their implementations of the data structures and libraries involved are easy to locate⁵.

Currently available versions of many popular applications are IP version independent, and support IPv6 in their default configurations. These include some of the most popular web browsers, web servers, email transfer agents, and domain name servers, among other applications that are already installed and in daily use on millions of home and office computers. Patches are also available on the web for several pieces of software that do not support IPv6 by default.

In short, after a network supports IPv6, there should be little or no difficulty in finding applications that will run over it using IPv6. Many of these are already likely to be installed and in use with IPv4. Since these applications are version independent, fallback to IPv4 operations while IPv6 network configurations are being fine-tuned can be painless and fairly automatic.

As for the transport protocols below applications, TCP and UDP are the most commonly used transports, and both support use over IPv6 on popular operating systems. Newer transport layer protocols like SCTP and DCCP also have no issues preventing them from being used over IPv6. In operating systems that support IPv6 (discussed later in Section 4.2.2), it is common for all of the standard included transport protocols to support operation with both IPv6 and IPv4 addresses.

In light of the readily available guidance and code examples for writing applications that work with both IPv6 and IPv4, and the preponderance of version independent applications that are already installed and support IPv6, application transition fears regarding IPv6 do not appear to be founded in reality.

4.2. Maturity Level

In a number of technical discussions, we have encountered opinions that IPv6 is either ephemeral (not available or not supported), still a work-in-progress that is not yet ready for widespread use, or has no market support. We factor in all of these notions, and analyze them based on several specific aspects including documentation, software/hardware support, real-world usage statistics, and government policy directives.

⁵ For instance, documentation from SCO's UnixWare 7 Release 7.14, specifically discusses porting applications: http://ou800doc.caldera.com/en/SDK_netapi/sockC.PortIPv4applIPv6.html.

4.2.1. Protocol Documentation

In December of 1995, RFC 1883 was published as a Proposed Standard for IPv6 [47]. Three years later, in December of 1998, RFC 2460 was published as a Draft Standard [1], This basic specification that covers the IPv6 header format, required extension headers, fragmentation behavior, flow labels, traffic classes, and upper-layer protocol issues has remained unchanged since its publication. Accompanying documents that can be considered the core of IPv6 include the specifications for ICMPv6 [48], Neighbor Discovery [16], and Address Autoconfiguration [11], all of which have reached the Draft Standard level as well. Many protocols that are well-accepted by industry and in widespread use are only formally published at the level below of Proposed Standards, and not at the level of maturity that IPv6 has attained in the IETF standards process. All of this demonstrates that the core IPv6 specification is agreed upon and stable, as has been the case for some time now.

Additionally, a very rough search on the rfc-editor.org website for “IPv6”, turns up 166 documents. For the most part, these document IPv6 usage and interactions in conjunction with other protocols, or extensions to IPv6. This search is fairly conservative in that a much larger number of documents deal with IPv6 in at least some way, but are not indexed under the term “IPv6” and so do not turn up. We simply use the large number of results that do show up as evidence that integration of IPv6 with numerous link layers and the extension of IPv6 have been actively pursued by industry, and a large number of supplementary standards have been produced. In April 2006, among IETF working groups, the “Mobility for IPv4” working group seemed to be the only one specifically chartered to provide an IPv4-only protocol, whereas at least 8 others were chartered specifically to provide IPv6-based solutions, including:

- IPv6 over Low-Power WPAN (6lowpan)
- IPv6 Working Group (ipv6)
- Mobility for IPv6 (mip6)
- Mobile IPv6 Signaling and Handoff Optimization (mipshop)
- Mobile Nodes and Multiple Interfaces in IPv6 (monami6)
- Site Multihoming by IPv6 Intermediation (shim6)
- Site Multihoming in IPv6 (multi6)
- IPv6 Operations (v6ops)

Among other working groups, for example those focusing on security, application, or transport protocols, it seems that the vast majority are constructing protocols that will work with IPv6 or both IPv6 and IPv4 (e.g. Host Identity Protocol [49]). There is a clear

sense of support for IPv6 in the standards community based on this survey of current IETF activities. From mailing list archives it can be seen that representatives from several large vendors and operators are active participants in the IPv6 groups, not merely academics.

In addition to the IETF, other bodies exist, such as the IPv6 Forum, to further the use and adoption of IPv6, and produce documentation and recommendations on the topic. There are widely available training courses and materials, textbooks, and support services for IPv6 deployment, transition, and troubleshooting. A search on amazon.com for IPv6 books turned up 60 results.

There is clearly a proliferation of materials and ongoing activities from both within and outside the IETF that serve as IPv6 resources, and serve as an indicator that IPv6 is supportable.

4.2.2. Running Code

Many IPv6 implementations are available from both commercial vendors and the open-source community. The IPv6 Forum has created the “IPv6 Ready” Logo Program, which consists of sets of criteria that can be used to assess the features and interoperability of IPv6 products. Phase-1 of this program judges implementations of basic or core IPv6 functions. The Phase-2 Logo builds upon Phase-1 by adding tests for IPsec and mobility features. As of August 1, 2006, 229 products had earned the Phase 1 IPv6 Ready Logo, and 46 products had earned the Phase 2 Logo. These products include some of the most popular router and end-host operating systems that are currently in use on millions of desktop, laptop, and palmtop computers as well as access, core, and border routers. Many products also contain more advanced IPv6 features beyond the basic sets that that IPv6 Ready program tests for.

Additionally, many common consumer devices come with IPv6. For instance, the 3GPP and 3GPP2 cellular telephony groups have made IPv6 a part of the IP Multimedia System (IMS). IPv6 capable, or dual-stack cellular handsets have been available for some time, and a dual-stack has been observed to take up only about 15% more space on these devices than an IPv4-only stack [50]. Some popular set-top boxes and home video game consoles are also IPv6-enabled.

4.2.3. Real-World Deployment

Here we examine evidence that the IPv6 Internet is currently operational. This evidence comes from four main sources (1) known IPv6 exchanges and peering services, (2) reports from the RIRs on IPv6 allocations, (3) BGP announcements, and (4)

measurements of 6-to-4 gateways. All of this evidence suggests that IPv6 is nearing a critical mass of operational use.

The www.v6nap.net web site lists 18 exchanges that support IPv6 throughout the United States, South Korea, the Netherlands, Finland, France, Germany, Japan, and the UK. As an example, all of the MAE exchange⁶ facilities are capable of exchanging IPv6 traffic. The same switches are used to support both IPv4 and IPv6, across a number of access types (ATN, Frame Relay, or Gigabit Ethernet). Native IPv6, dual-stack, and tunneled connections are supported at the customer's discretion. New IPv6-native exchange point addresses are automatically provided to customers with current IPv4 addresses at an exchange.

Additionally, a 2002 publication from PAIX indicated that IPv6 services have been incorporated there, and are in use by 10% of the customer base in Palo Alto, with growth expected [51]. Several major Japanese and Korean network providers are supporting IPv6 operationally. In some countries, IPv6 services are directly available to end-users through their ISPs⁷.

IPv6 adoption is rapidly accelerating as IPv6 infrastructure is deployed throughout much of the Internet backbone and major wide-area networks. Wide-area research and development networks have been running IPv6 infrastructure, services, and applications for several years. Across the globe, a number of major backbone network providers have deployed IPv6 services. CAIDA has published measurements of the IPv6 macroscopic topology and a geopolitical analysis of IPv6 deployment⁸.

IPv6 address blocks are assigned by IANA to the five RIRs. The RIRs then further distribute smaller blocks of addresses to IPv6 ISPs and other LIRs. Each of the five RIRs publishes some statistics on the prefixes that they have delegated, as well as the Autonomous System Numbers (ASNs) assigned (note that the percentage presented that compares the number of IPv6 prefixes to the number of ASNs is not a completely valid way to measure IPv6 adoption for a number of reasons):

- AfriNIC (April 24, 2006): 11 IPv6 prefixes (220 ASNs – 5 %)
- APNIC (April 24, 2006): 436 IPv6 prefixes (2162 ASNs – 20.2%)
- ARIN (April 24, 2006): 247 IPv6 prefixes (16729 ASNs – 1.5%)

⁶ More information can be found at <http://www.mae.net/>.

⁷ See “Life with IPv6” presentation slides from Keiichi Shima, February 2005, at: <http://member.wide.ad.jp/~shima/publications/20050216-ipv6-ipv6life-slides.pdf>.

⁸ See CAIDA's web pages at <http://www.caida.org/analysis/topology/macrosopic/IPv6> and <http://www.caida.org/analysis/geopolitical/bgp2country/ipv6.xml> for detailed data.

- LACNIC (April 21, 2006): 54 IPv6 prefixes (1060 ASNs – 5.1%)
- RIPE NCC (April 21, 2006): 761 IPv6 prefixes (11437 ASNs – 6.7%)

This data indicates that IPv6 addresses have been assigned to a fair number of LIRs, especially in the Asia-Pacific region. The current policies for IPv6 allocation from the RIRs do not allow address blocks to be assigned to end-sites (although this may be changed soon). This is not the case in IPv4, so the number of ASNs includes a number of IPv4 end-sites that are not eligible for IPv6 address blocks from an RIR, and thus the number of locations where IPv6 is usable is actually much greater than the percentages of prefixes over ASNs reported here. If Provider Independent addressing for IPv6 becomes popularly practiced among the RIRs, then we would expect these percentages to more accurately reflect the penetration of IPv6. Since ASNs may correspond to multiple prefixes, at full adoption, these would go somewhere above 100%. The percentage may also over-estimate in the case where networks have obtained IPv6 prefixes but not actually configured their routing protocols to advertise them.

In April of 2006, Geoff Huston's BGP analysis tool⁹ showed 721 active IPv6 BGP entries. Among these, 589 unique AS numbers appear, with 419 origin-only ASes, 12 transit-only ASes, and the remainder mixed. These BGP observations show that there is a global IPv6 routing table with a reasonable number of sites contained in it.

6-to-4 [52] is a transition mechanism that tunnels IPv6 over IPv4 packets for transit across IPv4-only portions of the Internet. Pekka Savola has studied the traffic at a public 6-to-4 gateway [53]. This was only considered to be a relatively small, or lightly used, 6-to-4 gateway, but it still was probed by 2 million Windows hosts, and actively used by over 1000 nodes per month in 2004. DNS, SSH, HTTP, SMTP, and BitTorrent file sharing traffic were all observed over the 6-to-4 gateway, indicating that typical Internet applications are functioning over it.

Fairly recent maps of the Internet showing IPv6 capabilities have been produced by Lumeta and are available on-line¹⁰.

4.2.4. Policy Directives

Among US government agencies, the Department of Defense (DoD) was an early recognizer of the benefits of IPv6 and began the deployment and transition process before most other federal agencies even considered using IPv6. The DoD has a number of useful resource publications for IPv6 including a set of feature profiles for judging

⁹ On the web at: <http://bgp.potaroo.net/v6/as6447>.

¹⁰ See <http://www.lumeta.com/IPv6>.

acquisitions against [54]. The DoD has announced plans to fully transition to IPv6 by fiscal year 2008.

In 2005, the Government Accountability Office (GAO) recommended to the Office of Management and Budget (OMB) that other federal agencies should follow DoD's lead and begin planning for a move to IPv6 [55]. Following this, it was announced that June 2008 was the deadline for all agencies to support IPv6 in their operational networks [56].

Plans for the 2008 Olympics in China involve IPv6 as a prominent means of connecting millions of users to various types of multimedia content [57]. In general, the growth in the "online population" in Asian countries is causing IPv6 to be eagerly deployed there.

Since several of the features of IPv6 can be back-ported or hacked into the IPv4 architecture through various means, IPv6 has been portrayed as unnecessary or lacking a killer-application by many pundits in the US. These opinions are not well-informed from the standpoint of network architecture, where attempting to make IPv4 do things that it was not designed for makes the network more fragile. For instance, the use of NAT to get around addressing limitations in IPv4 is well-known to have poor architectural implications. Unfortunately, US businesses still seem to be stalling on IPv6 deployment, although the recent government action in this area may serve to also motivate the private section to some extent.

4.3. Header Overhead

On many types of links currently in use, the capacity is often limited with respect to the demand. This includes wireless data links for cellular telephony, satellite communications, aeronautical traffic control and operations, and near and deep-space exploration. Operators and users of such links have sometimes expressed fears that enabling IPv6 would exacerbate the capacity-demand mismatch on their links, as basic IPv6 headers are larger than IPv4 headers and some of the alternative internetworking protocols that are in use in such domains¹¹.

For these types of links, several header compression schemes have been developed to reduce the sizes of IPv6 and IPv4 headers on the link [61], sometimes in conjunction with higher layer transport and security protocol headers [62]. In general, IPv6 headers can be compressed as well, or better than IPv4 headers¹². The IETF's Robust Header

¹¹ Examples include ATN for aeronautical communications based on ISO/OSI's CLNP [58], CCSDS Space Packet Protocol [59] and CCSDS SCPS-NP [60] for satellites and space-exploration.

¹² For instance, the lack of a checksum in IPv6 headers removes 16 bits of entropy that are present in IPv4 headers.

Compression working group has delivered particularly effective header compression solutions for IPv6 and IPv4 [63]. In most types of capacity-challenged links, header compression techniques are already in use, so given the close relative performance of IPv6 and IPv4 header compression techniques, there should be little to no impact on the link loading when IPv6 is used with appropriate header compression mechanisms.

In general terrestrial network links (for example, high-speed LANs and provider backbones), congestion is not a problem. Based on observations that the majority of the bytes transferred belong to packets larger than 500 bytes, with many around 1500 bytes [64], an additional 20 bytes per packet¹³ should be insignificant, in these cases.

A more detailed analysis of IPv6 versus IPv4 header overhead including the various IP options and extension headers used in a number of scenarios has been conducted by Eddy [65] and includes the following results:

- Given the maximum-sized payload possible in IPv4 (a 65,515 byte total packet), the base IPv4 header overhead is 0.03% of the packet's size, while IPv6's base header overhead is 0.06% of the packet's size. This difference is insignificant in any case.
- On the small end of payloads typical for bulk-transfer (generally constituting the majority of bytes on a link), with 1280 byte packets¹⁴, IPv4 base headers are 1.58% of the packet, while IPv6 headers are 3.13% of the packet. An increase of only a few percent in link loading should be insignificant in currently well-provisioned networks.
- In a simple example where fragmentation of a packet with a 1400 byte payload was required, the IPv4 headers generated had overhead of 2.78% of the generated packets on the affected links, while the IPv6 headers had overhead of 6.42% of the generated packets on the path. This comparison is less direct due to the end-to-end nature of IPv6 fragmentation, but in any case, the overhead for IPv6 is within the same order of magnitude as that for IPv4.
- In the case of jumbograms, supported by IPv6 but not by IPv4, the header overhead is vanishingly low. Since IPv4 does not support jumbograms, the header overhead and processing power needed to transmit and receive the equivalent number of IPv4 packets is somewhat higher than that for IPv6.

¹³ Based on a 40 byte IPv6 base header compared to a 20 byte IPv4 base header.

¹⁴ This is the minimum MTU that a link's sub-network convergence protocol must support for IPv6 packet transport.

- For mobility scenarios, direct comparisons are difficult due to the support of route-optimization in Mobile IPv6, but not Mobile IPv4, and the header overhead may vary based on the directionality of data transfer and a node's location either in its home network or a foreign network. In any case of mobility, with typical packets of 1280 total bytes being sent, the IPv6 and IPv4 headers do not exceed 7% of the packets sent.

It appears that in cases where header compression is used, using IPv6 implies equivalent link loading to using IPv4, whereas when header compression is not enabled, an increase on the order of 2-5% in link capacity might be observed. This analysis is based on naive assumptions of mostly bulk traffic patterns and does not include considerations for the differences in routing and configuration protocols (ARP/ND, DHCP, IGMPv3/MLDv2, etc). Based on anecdotes and a search of available materials, we did not find evidence that operators had observed large spikes in link utilization after enabling or upgrading to IPv6.

5. Conclusions / Summary

In conclusion, IPv6 offers many potential business case advantages over IPv4 and is currently possible to use successfully in production environments with readily available materials, possibly without even requiring hardware or software upgrades from currently used systems.

Particular aspects of IPv6 that we have positively identified as advances over IPv4 include:

- IPv6 enables addressing architectures that scale well in terms of the number of nodes and subnetworks, the size of subnetworks, and the degree of change within subnetworks; including practical cases where IPv4 becomes difficult to use robustly.
 - This is enabled by a number of factors in addition to simply longer addresses, including better architected address allocation policies, host and router autoconfiguration capabilities, and inclusion of scoped multicast and anycast in the base protocol.
- Global routing tables in IPv6 are potentially much simpler than their IPv4 counterparts, and thus require lower memory and computational resources.
 - This benefit can be realized assuming the reluctance to assign Provider Independent prefixes prevails. This might also result in less routing protocol traffic being transferred between peers as well as speeding the rate of convergence after topology disruptions.

- In resource-constrained environments, IPv6 requires less processing than IPv4, which can result in reduced power demands and latencies, especially for routers.
 - Stemming from the removal of the IPv4 checksum, the liberation of routers from fragmentation responsibilities, the ability to use the flow label rather than more intensive packet inspection, and the use of scoped multicast instead of broadcast, it seems that computational demands on both IPv6 routers and end-hosts are reduced in comparison to IPv4.
- The flow-label in IPv6 is an enabler for per-flow Quality of Service with simpler algorithms and more efficient implementations that also permit the remainder of a packet to be encrypted; all of which are precluded in IPv4.
- Network and device security is boosted in IPv6 based on address manipulation techniques and secure neighbor discovery features that have no IPv4 counterparts.
 - Security mechanisms such as cryptographically-generated addresses and address privacy capabilities are not possible to back-port onto IPv4 as was done with the IPsec protocols and some other features, due to the fact that these rely on the large difference between the number of bits in an IPv6 address and the number of bits required to uniquely identify each host and/or subnet within a site.
- Routing for mobile nodes is more efficient in IPv6 than in IPv4. Smooth handover techniques for IPv6 also exist with no IPv4 equivalents.
 - Partially due to the IPv6 improvements in extensibility, scalability, efficiency, and security, IPv6 has often been preferred for research involving mobile devices. The FMIPv6, HMIPv6, NEMO, and MONAMI6 extensions do not yet have equivalents for IPv4. Although in some cases approximations are possible, interest in IPv4 mobility has lagged in comparison to IPv6 mobility within the IETF.
- Current standards activities indicate that many future features may be developed for IPv6, but not necessarily for IPv4.

We examined a number of the fears that architects and engineers who may be moderately knowledgeable about IPv6, but are not IPv6 experts, have commonly shared. We have found that each one of these seems to be based on hearsay and conjecture rather than reality:

- Common currently-installed operating systems, transport protocols, and applications that are in use at many IPv4-only sites will actually support IPv6

with no patching or upgrading required, or in some cases minor configuration changes.

- Porting old applications and writing new applications to work in both IPv4 and IPv6 environments, as available, is relatively straightforward. Many resources are available and the changes in the socket API are well documented.
- Major router vendors support IPv6 in their product lines, and have been shipping IPv6-capable and IPv6-enabled products for several years.
- Network providers world-wide are offering IPv6 services, although these have been slow to reach the edges of the network where home-users are located, and particularly slow to be adopted in the US.
 - Major backbones and research networks almost unanimously include IPv6 services, with some offering IPv6 exclusively.
- Recent policy announcements and the demand for IPv6 features for government use in large nations like China and the US make the acceleration of IPv6 usage and transition likely.
- For capacity-constrained links, common header compression techniques are at least as effective on IPv6 packets as they are on IPv4 packets. Without compression IPv6 does not significantly impact the loading of uncongested links.

6. Acknowledgements

The main content of this document is amalgamated from three IETF Internet-Drafts [65, 66, 67], produced as inputs to NASA's ongoing design of network-centric architectures supporting space exploration and aeronautical communications. The authors wish to acknowledge helpful comments and suggestions toward these Internet-Drafts from John Loughney, David Green, and the members of NASA's Space Communications Architecture Working Group, and C3I Communications Adapter teams.

7. NAv6TF Disclaimer

Data and information is provided for informational purposes only, and is not intended for business purposes. Neither IPv6 Forum/NAv6TF or its affiliates nor any of its data or content providers shall be liable for any errors in the content, or for any actions taken in reliance thereon. IPv6 Forum/NAv6TF shall not be liable for any damages or costs of any type arising out of or in any way connected with your use of the content published herein.

8. About NAv6TF

The North American IPv6 Task Force (NAv6TF) www.nav6tf.org is a sub-chapter of the IPv6 Forum www.ipv6forum.org dedicated to the advancement and propagation of IPv6 (Internet Protocol, version 6) in the North American continent. Comprised of individual members, rather than corporate sponsors, the NAv6TF mission is to provide technical leadership and innovative thought for the successful integration of IPv6 into all facets of networking and telecommunications infrastructure, present and future.

Through its continued facilitation of technical and business case white-papers, IPv6-centric conferences, IPv6 test and interoperability events, IPv6 deployment readiness guides, and collaboration with IPv6 task forces from around the globe, the NAv6TF will strive to be the guiding force for IPv6 adoption and readiness in the U.S. and Canada.

9. About the Authors

Wesley M. Eddy works for Verizon Federal Network Systems as an on-site contractor at NASA's Glenn Research Center. His work focuses on protocol evaluation and development for use in space exploration and aeronautical applications.

William Ivancic is a senior research engineer at NASA's Glenn Research Center. He is a technical director of hybrid satellite/terrestrial networking, space-based Internet, and aeronautical Internet research, and is interested in large-scale, secure deployment of mobile networks. He is also a principal of Syzygy Engineering, a consulting company specializing in communications systems and networking as well as advanced technology risk assessment.

Joseph Ishac is a computer engineer at the NASA Glenn Research Center. His research specializes in networking protocols, and he designs and analyzes networked communications systems for space exploration and aeronautics.

10. References

- [1] Deering, S. and R. Hinden, "[Internet Protocol, Version 6 \(IPv6\) Specification](#)", RFC 2460, December 1998.
- [2] Postel, J. "[Internet Protocol](#)", STD 5, RFC 791, September 1981.
- [3] IANA, "[Special-Use IPv4 Addresses](#)", RFC 3330, September 2002.
- [4] Hinden, R., "[Applicability Statement for the Implementation of Classless Inter-Domain Routing \(CIDR\)](#)", RFC 1517, September 1993.
- [5] Rekhter, Y. and T. Li, "[An Architecture for IP Address Allocation with CIDR](#)", RFC 1518, September 1993.

- [6] Hubbard, K., Kouters, M. Conrad, D., Karrenberg, D., and J. Postel, "[INTERNET REGISTRY IP ALLOCATION GUIDELINES](#)", BCP 12, RFC 2050, November 1996.
- [7] Fuller, V. Li, T., Yu, J., and K. Varadan, "[Classless Inter-Domain Routing \(CIDR\): an Address Assignment and Aggregation Strategy](#)", RFC 1519, September 1993.
- [8] Hain, T., "[Architectural Implications of NAT](#)", RFC 2993, November 2000.
- [9] Holdrege, M. and P. Srisuresh, "[Protocol Complications with the IP Network Address Translator](#)", RFC3027, January 2001.
- [10] Hinden, R. and S. Deering, "[IP Version 6 Addressing Architecture](#)", RFC 4291, February 2006.
- [11] Thomson, S. and T. Narten. "[IPv6 Stateless Address Autoconfiguration](#)", RFC 2462, December 1998.
- [12] Miyakawa, S. and R. Droms, "[Requirements for IPv6 Prefix Delegation](#)", RFC 3769, June 2004.
- [13] Rijssinghani, A., "[Computation of the Internet Checksum via Incremental Update](#)", RFC 1624, May 1994.
- [14] Touch, J. and B. Parham, "[Implementing the Internet Checksum in Hardware](#)", RFC 1936, April 1996.
- [15] Plummer, D., "[Ethernet Address Resolution Protocol: Or converting network protocol addresses to 48.bit Ethernet address for transmission on Ethernet hardware](#)", STD 37, RFC 826, November 1982.
- [16] Narten, T., Nordmark, E., and W. Simpson, "[Neighbor Discovery for IP Version 6 \(IPv6\)](#)", RFC 2461, December 1998.
- [17] Nichols, K., Blake, S., Baker, F., and D. Black, "[Definition of the Differentiated Services Field \(DS Field\) in the IPv4 and IPv6 Headers](#)", RFC 2474, December 1998.
- [18] Partridge, C., "[Using the Flow Label Field in IPv6](#)", RFC 1809, June 1995.
- [19] Kent, S. and K. Seo, "[Security Architecture for the Internet Protocol](#)", RFC 4301, December 2005.
- [20] Dierks, T. and E. Rescorla, "[The Transport Layer Security \(TLS\) Protocol Version 1.1](#)", RFC 4346, April 2006.
- [21] Lynn, C., Kent, S., and K. Seo, "[X.509 Extensions for IP Addresses and AS Identifiers](#)", RFC 3779, June 2004.

- [22] Atkinson, R. "[Security Architecture for the Internet Protocol](#)", RFC 1825, August 1995.
- [23] Metzger, P. and W. Simpson, "[IP Authentication using Keyed MD5](#)", RFC 1828, August 1995.
- [24] Loughney, J., "[IPv6 Node Requirements](#)", RFC 4294, April 2006.
- [25] De Velde, G., Hain, T., Droms, R., Carpenter, B., and E. Klein, "[IPv6 Network Architecture Protection](#)", draft-ietf-v6ops-nap-02, Internet-Draft (work in progress), October 2005.
- [26] Narten, T. and R. Draves, "[Privacy Extensions for Stateless Address Autoconfiguration in IPv6](#)", RFC 3041, January 2001.
- [27] Arkko, J., Kempf, J., Zill, B., and P. Nikander, "[SEcure Neighbor Discovery \(SEND\)](#)", RFC 3971, March 2005.
- [28] Aura, T., "[Cryptographically Generated Addresses \(CGA\)](#)", RFC 3972, March 2005.
- [29] Merike Kaeo, David Green, Jim Bound, and Yanick Pouffary, "[IPv6 Security Technology Paper](#)", Version 1.0, North American IPv6 Task Force (NAv6TF) Technology Report, July 2006.
- [30] Perkins, C., "[IP Mobility Support for IPv4](#)", RFC 3344, August 2002.
- [31] Johnson, D., Perkins, C., and J. Arkko, "[Mobility Support in IPv6](#)", RFC 3775, June 2004.
- [32] Koodli, R. "[Fast Handovers for Mobile IPv6](#)", RFC 4068, July 2005.
- [33] Soliman, H., Castelluccia, C., El Malki, K., Bellier, L., "[Hierarchical Mobile IPv6 Mobility Management \(HMIPv6\)](#)", RFC 4140, August 2005.
- [34] Devarapalli, V., Wakikawa, R., Petrescu, A., and P. Thubert, "[Network Mobility \(NEMO\) Basic Support Protocol](#)", RFC 3963, January 2005.
- [35] Wakikawa, R., Ernst, T., Nagami, K., "[Multiple Care-of Addresses Registration](#)", draft-ietf-monami6-multiplecoa-00, Internet-Draft (work in progress), June 2006.
- [36] Cain, B., Deering, S., Kouvelas, I., Fenner, B., and A. Thyagarajan, "[Internet Group Management Protocol, Version 3](#)", RFC 3376, October 2002.
- [37] Vida, R. and L. Costa, "[Multicast Listener Discovery Version 2 \(MLDv2\) for IPv6](#)", RFC 3810, June 2004.
- [38] Partridge, C., Mendez, T., and W. Milliken, "[Host Anycasting Service](#)", RFC 1546, November 1993.

- [39] Woodcock, B., "[Best Practices in IPv4 Anycast Routing](#)", presentation slides version 0.9, August 2002.
- [40] Weber, S. and L. Cheng, "[A Survey of Anycast in IPv6 Networks](#)", IEEE Communications Magazine, January 2004.
- [41] Chown, T., "[Things to Think About When Renumbering an IPv6 Network](#)", draft-chown-v6ops-renumber-thinkabout-00, Internet-Draft (expired), October 2004.
- [42] Rosenberg, J., Weinberger, J., Huitema, C., and R. Mahy, "[STUN – Simple Traversal of User Datagram Protocol \(UDP\) Through Network Address Translators \(NATs\)](#)", RFC 3489, March 2003.
- [43] Huitema, C., "[Teredo: Tunneling IPv6 over UDP through Network Address Translators \(NATs\)](#)", RFC 4380, February 2006.
- [44] Shin, M-K., Hong, Y-G., Hagino, J., Savola, P., and E. M. Castro, "[Application Aspects of IPv6 Transition](#)", RFC 4038, March 2005.
- [45] Gilligan, R., Thomson, S., Bound, J., McCann, J., and W. Stevens, "[Basic Socket Interface Extensions for IPv6](#)", RFC 3493, February 2003.
- [46] Stevens, W., Thomas, M., Nordmark, E., and T. Jinmei, "[Advanced Sockets Application Program Interface \(API\) for IPv6](#)", RFC 3542, May 2003.
- [47] Deering, S. and R. Hinden, "[Internet Protocol, Version 6 \(IPv6\) Specification](#)", RFC 1883, December 1996.
- [48] Conta, A. and S. Deering, "[Internet Control Message Protocol \(ICMPv6\) for the Internet Protocol Version 6 \(IPv6\) Specification](#)", RFC 2463, December 1998.
- [49] Moskowitz, R. and P. Nikander, "[Host Identity Protocol \(HIP\) Architecture](#)", RFC 4423, May 2006.
- [50] Loughney, J., "[IPv6 in 2G and 3G Networks](#)", North American IPv6 Summit 2004, June 2004.
- [51] PAIX, "[IPv6: The Next Generation Internet Protocol](#)", NAv6TF Resources White Paper, November 2002.
- [52] Carpenter, B. and K. Moore, "[Connection of IPv6 Domains via IPv4 Clouds](#)", RFC 3056, February 2001.
- [53] Savola, P., "[Observations of IPv6 Traffic on a 6to4 Relay](#)", ACM Computer Communications Review, Volume 35, Number 1, January 2005.
- [54] Green, D., "[DoD IPv6 Standard Profiles for IPv6 Capable Products](#)", DISA draft for coordination, Draft v.06, December 2005.

- [55] Evans, K., "[Transition Planning for Internet Protocol Version 6](#)", Office of Management and Budget, Memorandum for the Chief Information Officers M-05-22, August 2005.
- [56] GAO, "[Federal Agencies Need to Plan for Transition and Manage Security Risks](#)", GAO report to congressional requesters GAO-05-471, May 2005.
- [57] Chi-Loong, C., "[China's IT Gold](#)", CMPnetAsia Newsletter, December 2005.
- [58] International Civil Aviation Organization, "[Manual of Technical Provisions for the Aeronautical Telecommunications Network \(ATN\)](#)", ICAO DOC 9705/AN956, November 2001.
- [59] CCSDS, "[Space Packet Protocol](#)", CCSDS 133.0-B-1, September 2003.
- [60] CCSDS, "[SCPS Network Protocol \(SCPS-NP\)](#)", CCSDS 713.0-B-1, May 1999.
- [61] Ishac, J., "[Survey of Header Compression Techniques](#)", NASA Glenn Research Center Technical Report TM-2001-211154, September 2001.
- [62] Bormann, C., Burmeister, C., Degermark, M., Fukushima, H., Hannu, H., Jonsson, L-E., Hakenberg, R., Koren, T., Le, K., Liu, Z., Martensson, A., Miyazaki, A., Svanbro, K., Wiebke, T., Yoshimura, T., and H. Zheng, "[RObust Header Compression \(ROHC\): Framework and four profiles: RTP, UDP, ESP, and uncompressed](#)", RFC 3095, July 2001.
- [63] Jonsson, L-E. and G. Pelletier, "[RObust Header Compression \(ROHC\): A Compression Profile for IP](#)", RFC 3843, June 2004.
- [64] Claffy, K., Miller, G. and K. Thompson, "[The Nature of the Beast: Recent Traffic Measurements from an Internet Backbone](#)", International Networking Conference (INET), April 1998.
- [65] Eddy, W., "[Comparison of IPv4 and IPv6 Header Overhead](#)", draft-eddy-ipv6-overhead-00, Internet-Draft (work in progress), May 2006.
- [66] Eddy, W. and W. Ivancic, "[Assessment of IPv6 Maturity](#)", Internet-Draft (work in progress), May 2006.
- [67] Eddy, W. and J. Ishac, "[Comparison of IPv6 and IPv4 Features](#)", draft-eddy-ipv6-ip4-comparison, Internet-Draft (work in progress), May 2006.