Network Working Group W. Eddy
Internet-Draft Verizon Federal Network Systems
Expires: November 10, 2006 J. Ishac
NASA Glenn Research Center
May 9, 2006

Comparison of IPv6 and IPv4 Features
draft-eddy-ipv6-ip4-comparison-00

Status of this Memo

Copyright Notice

Abstract

This document collects and comments on several aspects IPv6 that differ from IPv4, and what practical impacts these differences might have to an organization. This data can be used in decision-making processes where the business case for deploying IPv6 is under consideration.

Table of Contents

1. Introduction

While IPv4 has achieved widespread use and acclaim, its intended successor, IPv6, is still facing some hurdles in large-scale deployment. In both aeronautical networking and space networks a move towards network-centric operations and away from application-specific point-to-point links is occuring. In multiple groups that are attempting to define aeronautical or space networking architectures, the use of Internet protocols is well-accepted, but there is considerable uncertainty on whether to use IPv4, IPv6, or a dual-stack.

It is the technical opinion of many that IPv6 is favorable, due to some of its features (mobility and security are particularly important for network-centric operations). However, some decision-makers have been misinformed that IPv6 is equivalent to IPv4 with larger addresses. This document sheds light on the reality that IPv6 contains several additional features which may give it an improved business case over IPv4. Companion documents debunk arguments where IPv4 is brought forward as a favored choice based on the logic that it has lower "overhead" than IPv6 [Eddy06], and provide evidence that IPv6 is currently mature enough for widespread use [EI06].

This document is broken down as follows. Section 2 compares the addressing and routing functions and capabilities in IPv4 and IPv6. Section 3 describes Quality of Service (QoS) features in both protocols. Section 5 considers the mobility extensions to each protocol. Section 6 discusses the multicast and anycast capabilities of IPv4 and IPv6, and Section 7 examines flexibility and extensibility.

2. Addressing and Routing

The most obvious difference between IPv4 and IPv6 is the change in the address size from 32 bits in IPv4 [RFC0791] to 128 bits in IPv6 [RFC2460]. 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, however, the actual contrast is less direct than this simple factor.

Aside from a few specific blocks for local-use, multicast, or other specific functions, the majority of the IPv4 32-bit address space is designated for global unicast addresses [RFC3330]. In the IPv4 addressing architecture, IANA delegates Regional Internet Registries (RIRs) /8 address blocks (8-bit network identifiers), which the RIRs can then divide into variable-length blocks for further assignment to ISPs or other registries [RFC2050] [RFC1519]. The maximum address block that a site could ever be given is a /8, which leaves only 24-bits for subnetting and addressing within the organization. Historically, large or complexly-organized groups 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. To compound matters, even using multiple /8s is a poor solution, since there is no guarantee that they 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. In recent years, many IPv4 users have circumented these issues by using Network Address Translators (NATs), although this practice is known to be frought with problems of its own [RFC2993] [RFC3027].

According to the IPv6 addressing architecture [RFC4291], the prefix of 001 identifies IPv6 global unicast addresses, so 1/8 of the address space, or 2^125 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 LIRs is that each LIR receive a minimum of a /32. The minimum sized address block that an LIR can then give to a site is a /48. For more details see: http://www.iana.org/ipaddress/ipv6-allocation-policy-26jun02. Since an IPv6 site can expect a *minimum* of a /48, this gives 16 bits of subneting space and 64 bits of interface identifier within a subnet (80 bits combined). Contrast this to an IPv4 site that can expect a *maximum* of a /8, leaving only 24 bits of space to be used for subnetting and host addressing combined. Since in reality, most sites do not get /8s, but rather /16s or /24s, there are more likely to be only 4-8 bits for subneting and 4-8 bits for identifying hosts within a subnet.

The IPv6 addressing architecture includes 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, which allow a node to find or communicate with routers without knowing their unicast address ahead of time. In IPv4, this is not possible without the assistance of other protocols.

Configuring and managing addresses in IPv6 and IPv4 can both be accomplished using versions of the DHCP protocol. However, IPv6 has mechanisms that allow a node to configure its own globally routable address, without the need for DHCP [RFC2462], 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. 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 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 idea is that the global routing table only has 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 addresses. This prevents routers from using large amounts of memory on the routing table, thus allowing lookups to be faster, and network operators to spend less money on expensive router memory upgrades. Recent developments indicate that Provider Independent addressing may become more prevalent in IPv6 assignments, and so this feature could be negated.

In the effort to build faster router platforms, two well-known speed-bumps in IPv4 were performing the checksum operations, and fragmenting datagrams, when required. While relatively efficient means of computing the IPv4 checksum [RFC1624], and even implementing it in hardware [RFC1936], were developed, it was decided to improve speed by not including any checksum at all in IPv6. The rationale behind this is that most link-layer protocols have at least their own checksum (and often their own retransmission protocols and error-correcting codes), and reliable application or transport protocols

also implement checksums. Since some of the link and higher-layer checksums in use were actually more powerful than the simple IPv4 checksum, it was 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" when fragmentation is required. This can represent a bottleneck that limits throughput and loads the central processor (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 weakness. In IPv6, routers never fragment packets; packets larger than an outgoing link's MTU are dropped. It is a source node's duty to pro-actively fragment its own packets. The lack of checksum and fragmentation responsibilties potentially allow IPv6 routers to perform slightly faster and with lower power requirements, but these differences are likely to be fairly minimal under typical use cases.

Another difference between IPv4 and IPv6 is in the way that IP addresses within a subnet are resolved into link-layer protocol addresses. IPv4 used the ARP [RFC0826] mechanism for this, while IPv6 uses Neighbor Discovery (ND) [RFC2461]. There are at least two key differences betwen 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 identifying ND and IPv6 payload types do not have to be defined, whereas IPv4 and ARP require separate codepoints. This is of only marginal importance. The main difference between ARP and ND, is that IPv6's ND is highly extensible and 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 the MTU. IPv4's ARP has no such facilities and no means for extension.

Altogether, in terms of addressing, routing, and forwarding features, IPv6 has advantages over IPv4 in every respect considered.

3. 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 [RFC2474]. In this respect, IPv4 and IPv6 are equivalent.

A significant capability that is part of the standard IPv6 header, and is not present in IPv4, is the ability to classify traffic into flows based on a flow label header field. This can be used as a basic building block to efficiently support QoS policies and protocols. In IPv4, flows can only be classified by the relatively expensive process of examining (and possibly parsing) header fields. Thus, certain types of per-flow QoS can be enabled in routers with lower computational overhead when using IPv6.

4. Security

Both IPv4 and IPv6 can be used in conjunction with the IPsec suite of protocols [RFC4301]. 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 [RFC4346] runs over top of the transport layer, and does not interface directly with IP, it is similarly mostly agnostic to the version of IP that is used. Both TCP and SCTP can run TLS, and both will run over IPv4 and IPv6. Additionally, the X.509 format for certificates that is often used in IPsec and TLS, has encoding methods for both IPv4 and IPv6 addresses [RFC3779]. So, the two most prevalent security architectures in the Internet suite, IPsec and TLS, have no significant differences between use with IPv4 and IPv6.

It is often touted that IPv6 has superior security properties to IPv4. In the majority of cases, the reasoning used to justify this is that IPsec is a part of IPv6, because in early IPsec specifications [RFC1825], it was stated that all IPv6-capable hosts MUST implement the IPsec Authentication Header in a basic configuration (keyed-MD5 with 128-bit key [RFC1828]), while for IPv4 supporting any part of IPsec was optional. In fact, current IPv6 node requirements mandate that IPv6 nodes MUST support both Authentication Header and Encapsulating Security Payload portions of IPsec [RFC4294]. However, since IPv6's core functions do not rely on IPsec, and only support for manual keying is required, 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 can more certainly be expected 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 [VHDCK05] that are relevent 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 things like easily being able to correllate remote log files of world-wide web activity [RFC3041]. IPv4 has no equivalent capability. 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 similar technique

that involves changing the prefix refering to a subnet, rather than the interface identifier. This prevents an attacker from being able to easily determine other related addresses from a known address.

In addition, IPv6 has the optional secure neighbor discovery extension, which allows hosts to authenticate the ND messages [RFC3971], and uses cryptographically-generated addresses to prove address ownership without any certificate management or other security infrastructure [RFC3972]. IPv4 has no comparable features, and its address space is too small for the features to be portable to IPv4.

In summary, IPv6 does have superior security features 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 version.

5. Mobility

Support for node mobility is not required in either IPv4 or IPv6, however, both protocols support mobility extensions [RFC3344] [RFC3775]. The means of supporting mobility, and the features of each mobility protocol differ. Mobile IPv4 uses UDP for signaling, whereas Mobile IPv6 uses IPv6 extension headers. This allows for a cleaner implementation, since the code can 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. 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 routing indirection that Mobile IPv4 depends on for reaching mobile nodes. In certain business scenarios, the Mobile IPv6 route optimization feature might actually result in saving money, since it greatly reduces the amount of traffic to and from the home network.

Mobile routers (and correspondingly, the mobile networks behind them) are supported in Mobile IPv4, but their operation is not particularly well specified. In contrast, Mobile IPv6 mobile routers, called NEMO routers, have been specified very clearly in their own standards documents [RFC3963]. 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, and not in the context of Mobile IPv4.

In summary, based on cleaner design, support for route optimization, and NEMO extensions, IPv6 has superior mobility features than IPv4.

6. Multicast and Anycast

IPv4 and IPv6 are both capable of supporting network-layer multicast communications. The major differences between IPv4 and IPv6 in terms of multicast lie mainly in the fact that multicast support is considered an "additional" 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, along with scopes for interface-local, admin-local, site-local, organization-local, and global) [RFC4291]. This is used as a building block for the ND service for autoconfiguration in IPv6. Broadcast addresses as used in IPv4 are then replaced with multicast addresss of the appropriate scope.

The basic host to router multicast protocol in IPv4 is IGMPv3 [RFC3376]. In IPv6, this function is filled by MLDv2 [RFC3810], which is functionally equivalent. 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 inside IPv6. The difference is in the reuse of ICMPv6 as a general purpose control messaging/signaling protocol, rather than defining a new protocol number with separate processing.

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 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 multicast to perform 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 mapping between interfaces and IPv4 addresses. Since IPv6 specifically supports assigning several addresses to an interface, multicast support is more straightforward.

While the early work on the concept of anycast involved IPv4 [RFC1546], and in the IPv4 Internet, anycast is actively being used in particular niches [Woodcock02], anycast features are not formally a feature of the IPv4 standard, but are supported 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 multihomed sites. However, due to the difference 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. Particularly it is not clear how IPv6 anycast will be used on a global scale [WC04]. 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 IPv4 versus IPv6 anycast features at this time.

7. Flexibility and Growth

Enterprise network designers have a strong desire for their networks to be able to grow with an organizations 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 [Chown04].

As noted in the companion IPv6 maturity study [EI06], there is currently only one IETF working group that seems to be chartered to provide an IPv4-specific solution, while there are many groups working on IPv6-specific solutions. This indicates that in the future, it is possible that a number of specific network layer enhancements may only be available for IPv6 networks. 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 [RFC2993] [RFC3027]. In IPv6, the common NAT functionalities that network administrators are interested in can all be performed without any of the negative repercussions [VHDCK05]. The ability to deploy new applications without any concern for application layer gateways in the NAT, or complex tunneling mechanisms [RFC3489] [RFC4380] alone is large practical benefit of IPv6.

8. Security Considerations

This informational document only contains informational text about IPv6 and IPv4 features. There are no new security considerations raised by this material.

9. Acknowledgements

Work on this document was performed at NASA's Glenn Research Center, in support of the NASA Space Communications Architecture Working Group (SCAWG), and the FAA/Eurocontrol Future Communications Study (FCS). Will Ivancic of NASA contributed useful comments on this document.

10. Informative References

[Chown04] Chown, T., "Things to Think About When Renumbering an IPv6 Network", draft-chown-v6ops-renumber-thinkabout-00 Internet-Draft (expired), October 2004.

[EI06] Eddy, W. and W. Ivancic, "Assessment of IPv6 Maturity", draft-eddy-ipv6-maturity-00 Internet-Draft (work in progress), May 2006.

[Eddy06] Eddy, W., "Comparison of IPv4 and IPv6 Header Overhead", draft-eddy-ip-overhead-00 Internet-Draft (work in progress), May 2006.

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981.

[RFC0826] 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.

[RFC1519] Fuller, V., Li, T., Yu, J., and K. Varadhan, "Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy", RFC 1519, September 1993.

[RFC1546] Partridge, C., Mendez, T., and W. Milliken, "Host Anycasting Service", RFC 1546, November 1993.

[RFC1624] Rijsinghani, A., "Computation of the Internet Checksum via Incremental Update", RFC 1624, May 1994.

[RFC1825] Atkinson, R., "Security Architecture for the Internet Protocol", RFC 1825, August 1995.

[RFC1828] Metzger, P. and W. Simpson, "IP Authentication using Keyed MD5", RFC 1828, August 1995.

[RFC1936] Touch, J. and B. Parham, "Implementing the Internet

Checksum in Hardware", RFC 1936, April 1996.

[RFC2050] Hubbard, K., Kosters, M., Conrad, D., Karrenberg, D., and J. Postel, "INTERNET REGISTRY IP ALLOCATION GUIDELINES", BCP 12, RFC 2050, November 1996.

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[RFC2461] Narten, T., Nordmark, E., and W. Simpson, "Neighbor Discovery for IP Version 6 (IPv6)", RFC 2461, December 1998.

[RFC2462] Thomson, S. and T. Narten, "IPv6 Stateless Address Autoconfiguration", RFC 2462, December 1998.

[RFC2474] 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.

[RFC2993] Hain, T., "Architectural Implications of NAT", RFC 2993, November 2000.

[RFC3027] Holdrege, M. and P. Srisuresh, "Protocol Complications with the IP Network Address Translator", RFC 3027, January 2001.

[RFC3041] Narten, T. and R. Draves, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", RFC 3041, January 2001.

[RFC3330] IANA, "Special-Use IPv4 Addresses", RFC 3330, September 2002.

[RFC3344] Perkins, C., "IP Mobility Support for IPv4", RFC 3344, August 2002.

[RFC3376] Cain, B., Deering, S., Kouvelas, I., Fenner, B., and A. Thyagarajan, "Internet Group Management Protocol, Version 3", RFC 3376, October 2002.

[RFC3489] 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.

[RFC3775] Johnson, D., Perkins, C., and J. Arkko, "Mobility Support

in IPv6", RFC 3775, June 2004.

[RFC3779] Lynn, C., Kent, S., and K. Seo, "X.509 Extensions for IP Addresses and AS Identifiers", RFC 3779, June 2004.

[RFC3810] Vida, R. and L. Costa, "Multicast Listener Discovery Version 2 (MLDv2) for IPv6", RFC 3810, June 2004.

[RFC3963] Devarapalli, V., Wakikawa, R., Petrescu, A., and P. Thubert, "Network Mobility (NEMO) Basic Support Protocol", RFC 3963, January 2005.

[RFC3971] Arkko, J., Kempf, J., Zill, B., and P. Nikander, "SEcure Neighbor Discovery (SEND)", RFC 3971, March 2005.

[RFC3972] Aura, T., "Cryptographically Generated Addresses (CGA)", RFC 3972, March 2005.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, February 2006.

[RFC4294] Loughney, J., "IPv6 Node Requirements", RFC 4294, April 2006.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.

[RFC4346] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.1", RFC 4346, April 2006.

[RFC4380] Huitema, C., "Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs)", RFC 4380, February 2006.

[RFC4489] Park, J-S., Shin, M-K., and H-J. Kim, "A Method for Generating Link-Scoped IPv6 Multicast Addresses", RFC 4489, April 2006.

[VHDCK05] 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.

[WC04] Weber, S. and L. Cheng, "A Survey of Anycast in IPv6 Networks", IEEE Communications Magazine , January 2004.

[Woodcock02]
Woodcock, B., "Best Practices in IPv4 Anycast Routing",

presentation slides version 0.9, August 2002.

Authors' Addresses

Wesley M. Eddy
Verizon Federal Network Systems
21000 Brookpark Rd, MS 54-5
Cleveland, OH 44135

Phone: 216-433-6682
Email: weddy@grc.nasa.gov

Joseph Ishac
NASA Glenn Research Center
21000 Brookpark Rd, MS 54-5
Cleveland, OH 44135

Phone: 216-433-3494
Email: jishac@grc.nasa.gov

Intellectual Property Statement

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.