Making Progress on IPv6: New Insights after an IPv4-IPv6 Dual-Stack Deployment

Shaun M. Lynch, Ph.D.
Department of Mathematics and Computer Science
University of Wisconsin-Superior
Superior, WI 54880
slynch@uwsuper.edu

Abstract

The Department of Mathematics and Computer Science upgraded its network to a full-fledged IPv4-IPv6 dual-stack infrastructure at the beginning of Fall Semester 2013. Three years later and after several refinements to the system, the author developed new insights into IPv6 that include strategies to deploy a dual-stack network, how to communicate and visualize IPv6 design patterns, ways to contrast essential differences between IPv6 and IPv4, and differences between the way various operating systems and networking applications implement IPv6 protocols. This paper tracks the author's journey from IPv4 to IPv6 while identifying conceptual barriers that may prevent individuals from fully utilizing the technology. A variety of topics are presented that focus on patterns of thinking, deployment schemes, and technology discrepancies that should be watched for. Examples are provided and suggestions offered to help guide those familiar with IPv4 technologies though the transition period.

1 Introduction

The Department of Mathematics and Computer Science upgraded its network to include a full-fledged IPv4-IPv6 dual-stack infrastructure at the beginning of Fall Semester 2013. One of the key motivations for deploying IPv6 was to enable faculty and staff to gain in-depth familiarity with IPv6 networking technologies and work through the challenge of deploying a dual-stack network. Three years later and after several refinements to the system, the author developed new insights into IPv6 that include strategies to deploy a dual-stack network, how to communicate and visualize IPv6 design patterns, ways to contrast essential differences between IPv6 and IPv4, and differences between the way various operating systems and networking applications implement IPv6 protocols.

During this period, the author observed that knowledge and familiarity with IPv4 form patterns of thinking that often works against students and practitioners transitioning to IPv6. For many, the most notable feature of IPv6 is the expanded address space provided by the 128 bit addressing scheme. This is a real consideration given the pressing issue of IPv4 address exhaustion; however, the strength of IPv6 lies in the new capabilities it brings such as auto-configuration; link-local, unique-local, and global addressing schemes; multicasting; and a variety of address assignment options. The challenge for educators is to communicate this improved functionality to current and future network architects and engineers in a meaningful and understandable manner that constructively draws upon prior knowledge.

This paper tracks the author's journey from IPv4 to IPv6 while identifying conceptual barriers that may prevent individuals from fully utilizing the technology. Several topics are addressed including: transitioning to IPv6, utilizing the IPv6 address model, embracing IPv6 private networks, configuring IPv6 host addresses, and revealing DHCPv6 service nuances. Examples are provided that illustrate how IPv6 can be configured to accommodate various design scenarios and attention is drawn to different approaches operating systems and network applications pursue when implementing IPv6. Finally, the paper closes with a comparison to another transitional period where a new technology supplants a previously established technology and the role instructors play in charting a path through the transition.

2 Transitioning to IPv6

A network revolves around a handful of essential parameters that include: 1) a network prefix that defines the range of host addresses, 2) a router that provides a gateway for the network, and 3) one or more directory services used to resolve named network resources. In addition, a mechanism must exist to assign network parameters to host systems whether manually using a static host configuration or automatically using some form of dynamic host configuration.

Connecting a client system to an IPv4 network is a rather straight forward task. Settings include an IP address that specifies the network prefix and host identification number, the

IP address of the router serving as the default gateway, and IP addresses of one or more domain name servers. While the IP address of the domain name servers may or may not be on the same network as the client, the client and default gateway must be the same network to ensure connectivity.

Although specifics may vary between operating systems and network applications, IT professionals benefit from the nearly universal manner this process is implemented and have come to rely on its consistency. If a client system is configured manually, these three settings are entered via graphical user interface, configuration file, or directly using the command line. If a client is to be configured automatically, these three parameters are entered in the scope and options settings of a DHCP (Dynamic Host Control Protocol) server.

Unfortunately, IPv4 was not designed with the ability to extend the physical address range beyond 32 bits thus inhibiting forward compatibility. In addition, there are other limitation that emerged as the protocol evolved to accommodate an increasing reliance on networked systems. IPv6 was developed to remedy these issues but in doing so, broke with the traditional system thus preventing it from being backward compatible.

Individuals familiar with IPv4 that are just beginning to dabble in IPv6 immediately stumble into problems due to the difference between the two protocols. A myriad of questions arise such as,

- How am I going to make sense of these 128 bit addresses?
- Why does my network adapter have multiple IPv6 addresses?
- What happened to NAT?
- How do I create a private network?
- Why can’t I automatically assign DNS addresses using router advertisement?
- Why can’t I assign a default gateway in DHCPv6?
- What happened to the MAC addresses in DHCPv6 leases and what are those DUID and IAID values?

Each question is valid when approaching the transition with an IPv4 mindset and answers are not necessarily forthcoming despite all the articles written on IPv6. Confusing the matter further, many operating systems and network applications present IPv6 interfaces that mimic the look and feel of their IPv4 counterpart thus perpetuating the problem. What is needed is a simple way to envision the configuration of an IPv6 network and a model that describes where specific functionality resides.

3 Utilizing the IPv6 Address Model

IPv6 is based on three important principles. First, auto-configuration is an essential part of the protocol. Second, the role of managing dynamic network settings has shifted from DHCP to router advertisement. Third, network functionality is defined by a collection of address spaces. For those transitioning to IPv6, auto-configuration can be a truly foreign concept in that nearly every setting is entered manually in IPv4. For instance, specifying

a default gateway in DHCPv4 entails looking up the router address and entering it as either a global or scope specific option. Although settings are centralized, manual entry is prone to error and requires intervention should the setting change. In IPv6, the key to successfully designing a network is to automate settings.

Automating network settings starts with configuring router advertisement. The basic function of router advertisement is to announce the IP address of the routing device. This is used by client devices to configure the default gateway parameter. Announcements are generated periodically for or on demand by clients that reside on the network link. In this case, a network link is defined by a collection of systems that can connect at the data link layer (Layer-2) of the OSI model. In IPv4, this is often called a broadcast domain. Setting up router advertising is left to the particular implementation, but it usually entails configuring a few basic settings and starting the service.

In IPv6, the router advertisement message includes additional flags and data that specify advanced client configuration options. For instance, by adding a network prefix and setting the autonomous flag, client systems can generate their own IP address using stateless address auto-configuration (SLAAC). By setting the managed and options flags, client systems can request an IP address and network settings from a DHCPv6 server using stateful address configuration. Router advertisement flags are often independent allowing clients to configure all the capabilities specified or to override particular flags and use a subset of capabilities [1].

For those transitioning from IPv4, it is important to recognize that IPv6 network configuration is not completely defined or constrained by a single address space. Instead, IPv6 defines multiple address spaces that have different routing characteristics. Address spaces in IPv6 can generally be divided into three categories that include non-routable, locally-routable, and globally-routable addresses that correspond with the link, site and internet, respectively. Together, these address spaces form tiers in which particular network functionality can be assigned as shown in Figure 1. Although deciphering the 128 bit address can be can be daunting at first, clues exist that allow one to quickly identify the network type and associate network resources for any given tier.

The foundation of IPv6 resides in the bottom-most layer of the diagram and is defined by the network link. Configuration of this tier is governed by a router hosting the router advertisement service. By default, all other devices connected to the network link will auto-configure their network parameters based on settings in the router advertisement message unless overridden. IP addresses in this tier are referred to as link-local addresses and can be identified by the prefix “fe80” (fe80::/10 in CIDR notation). Link-local addresses are non-routable and include a host identifier (e.g. fe80::*20c:29ff:febb:2957*) [2] [3]. The host identifier is created using a modified EUI-64 format that incorporates the MAC address of the network adapter [2]. Unfortunately, these host identifiers can be tracked so they may be privatized using a randomization scheme [4].

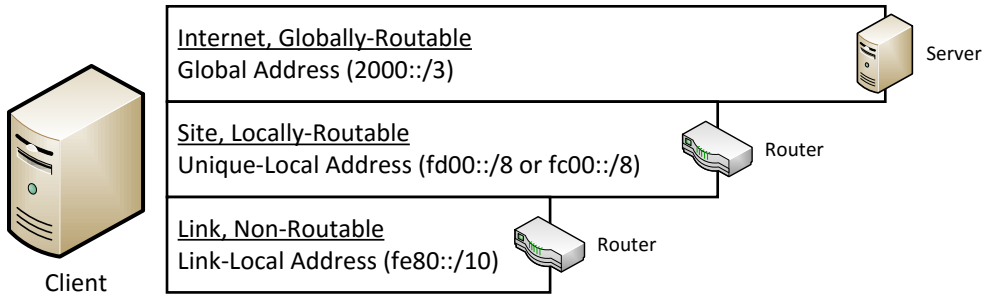

Figure 1: A model representing IPv6 as a collection of tiered address spaces

The middle layer of the diagram depicts the site tier. This tier is optional and includes any number of private networks utilizing unique-local addresses. A private network consists of one or more network links and allows data packets to be routed within the limits of the site. User workstations and local services generally reside in this tier. Examples include: Microsoft Active Directory and Domain Services, private DNS services, DHCP, update servers, database servers, application servers, and file servers. Unique-local addresses begin with the prefix "fc" or "fd" (fc00::/8 and fd00::/8 in CIDR notation, respectively) and are locally-routable [2] [3]. The network portion of the unique-local address includes a 40-bit network identifier along with a 16-bit subnet field. Unique-local addresses can be added to the network adapter manually, using SLAAC, or assigned by a DHCPv6 server. If used, the locally-routable tier extends to the internet-facing router that defines the perimeter of the site.

The top-most layer of the diagram illustrates the internet tier. This tier is necessary to achieve end-to-end connectivity between local and internet-based systems. In principle, only clients that require internet access such as internet-facing services should be placed in this tier. Examples include: public DNS servers, internet proxy servers, web server, authoritative time servers, and email servers. Global addresses begin with the prefix "2" (2000::/3 in CIDR notation) and can traverse all intermediate routers if permitted [2] [3]. The network portion of the global address includes a 48-bit routing prefix provided by the ISP plus a 16-bit subnet field. Like unique-local addresses, they can be added to the network adapter manually, using SLAAC, or assigned by a DHCPv6 server.

4 Embracing IPv6 Private Networks

Many making the transition to IPv6 quickly relate to the idea of global addresses since they have a counterpart in IPv4. When faced with the fact that IPv6 does not have provisions for network address translation (NAT) the question arises, "How do you create a private network?" In this case, it is best to first look at what NAT is and what is not.

Originally, NAT was designed as an address expansion scheme to avoid IPv4 address exhaustion [5]. To do this, a resident node serving as a router on a host network would map addresses to client systems on an adjoining network through the address of the resident node. When used in conjunction with IP masquerading, the addresses on the adjoining network are invisible to the host network thus creating what appears to be a

private network. However, clients within the adjoining network have unfettered access to the host network and beyond.

There is a major flaw to this concept of privacy that makes NAT less than suitable for a truly private network. The flaw lies at the asymmetric access facilitated by NAT that allows outbound traffic by default. Although getting access to a private network via the public network may be difficult, it is not the only way to gain access. There are a number of ways to gain access to a private network usually through other means such as social engineering, phishing attacks, downloading malware, physical entry, *et cetera*. Once a foothold is gained inside the private network, the default behavior is to let all traffic out. The only way to prevent this is to create rules that explicitly curtail outbound traffic.

Why is this important? A number of high profile attacks (i.e. Target-2013, Home Depot-2014, Office of Personnel Management-2015, *et cetera*) have involved the exfiltration of personally identifiable information, credit card numbers, trade-secrets, and other sensitive data. The attackers circumvented the NAT device and exploited the internal networks using alternate means. A disabled firewall or a flawed outbound rule would be sufficient to allow traffic to move offsite. In IPv6, private networks use unique-local addresses that are not internet routable. This means that all inbound and outbound traffic stops at the internet-facing router by default unless permission is granted. This greatly improves the organization's security posture since provisions must be made to explicitly allow access.

Internal private networks can gain access to internet resources indirectly using internet-facing devices located at the edge of the site network. Typically, these edge devices are specially designed, internet hardened devices that permit and track specific types of traffic. In many cases, these devices can be created around three basic services shown in Table 1. The servers hosting the services are assigned global addresses and would be located in the site usually within an isolated zone such as the DMZ. Because the servers are within the site, they can be accessed by other systems in the private network.

Service	Protocols/Ports	Device
Domain Name Services	domain/53	DNS Server/Appliance
Authoritative Time Services	ntp/123	NTP Time Server/Appliance
Web Services	http/80; https/443; ftp/20,21; etc.	Web/Internet Proxy

Table 1: List of edge device services, protocols, and devices.

5 Configuring IPv6 Host Addresses

In IPv4 networks, servers and network appliances are often configured manually using static settings while all other systems receive network settings from one or more DHCPv4 servers. In an IPv6 environment, network parameters may be set manually, generated locally through auto-configuration, or provided from an external source such as a router or DHCPv6 server. A typical system—client or server—may have network

settings from anyone or all the methods described! Although the default gateway parameter can be set manually, a properly configured network using router advertisement should provide the default route automatically. This gives the administrator the ability to configure the host IP address and DNS settings independently using any number of methods available.

A host IPv6 address can be set manually using a static setting, generated locally using the network prefix provided by router advertisement, or leased from a DHCPv6 server; whereas, DNS settings must be imported using DHCPv6 or through router advertisement. There are two important points to consider however. First, IPv6 allows for any number of IP addresses but it is often more effective to have only one unique-local or global address assigned in addition to the mandatory link-local address. Second, operating systems have significant differences in the manner in which they implement IPv6 leading to substantial inconsistencies across platforms [6] [7].

Assigning the correct network parameters entails carefully selecting and managing router advertisement flags and the IP setting on the network interface. Router advertisement flags provide signals to the client on how to configure its network parameters. Flags include [1]:

- Autonomous Address Configuration (A) – Signals the client to use stateless address auto-configuration (SLAAC) using the network prefix provided.
- Managed Address Configuration (M) – Signals that a DHCPv6 service is available and that the client should use stateful address auto-configuration.
- Other Configuration (O) – Indicates that additional network configuration is available from a DHCPv6 service.

While enabling or disabling router discovery, DHCPv6, and managed address configuration can be accomplished on the network interface. It is important to note two caveats however. First, it is a requirement that clients must implement SLAAC whereas implementing DHCPv6 is optional [8]. Therefore, disabling router discovery on the network interface is the only way to prevent SLAAC when the Autonomous Flag is set. Second, operating systems often attempt various methods to set networking parameters despite router advertising settings. This produces behavior that often contradicts the objectives of the intended configuration [7].

Router advertisement can also provide clients with DNS settings. It is important to note however, that not all operating systems recognize this feature. While many Linux distributions will assign DNS settings provided by router advertisement, clients running Windows will not. Therefore, a DHCPv6 server is a better choice for heterogeneous environments that include a mix of Windows and Linux operating systems [6].

Although there are a wide variety of host configuration options available, partitioning the network into manageable zones is still an essential part of network design. Distinguishing characteristics such as the type of function systems perform, the access systems should have to internet resources, the physical location of computing devices, and the policies deployed are issues that must be included in planning. Incorporating router advertisement

considerations into the planning process makes it much easier to deploy uniform settings without having to customize individual systems.

6 Revealing DHCPv6 Service Nuances

If a network environment is currently using DHCPv4 to manage host network settings, then it reasonable to consider using DHCPv6 to manage IPv6 clients. DHCP offers many advantages although arguably the most important is the ability to centralize network settings. There are however distinct differences between DHCPv4 and DHCPv6 that may cause confusion while making the transition.

Some differences are specific to individual operating systems. For example, the DHCPv4 service in Windows Server allows scope starting and ending addresses to be set along with exclusions that block additional ranges. Setting the scope in the DHCPv6 service entails entering the network prefix which automatically includes the entire host address range (e.g. 2001:db8::*1*-2001:db8::*ffff:ffff:ffff:ffff*) making for unwieldy host identifiers. The only way to limit the address range is to add exclusions that bound the desired range of host identifiers.

Another significant difference administrators notice in DHCPv6 is the much shorter list of global and scope options. The options available use a different numbering system and often have different names. Key options that set the IP addresses of DNS servers and domain name still exists, but options necessary for PXE boot are not immediately available. There are provision to add options although it is important to determine if the particular functionality is available and works as anticipated.

Unlike DHCPv4 where clients are identified by the MAC address of the network adapter, DHCPv6 uses a combination of identifiers called the DHCP Unique Identifier (DUID) and Interface Adapter Identifier (IAID) [9]. The DUID and IAID were introduced to uniquely identify the host and the specific network adapter, respectively. It is important to note that the DUID is not associated with anything physical; instead, it is created by the operating systems during installation. The DUID and IAID are both required to create lease reservations in DHCPv6.

7 Concluding Remarks

The wide deployment and staying power of IPv4 is a testament to its success. In many ways, this makes the transition to IPv6 more challenging since it requires IT professionals relearn basic tenets. In reality, the differences revolve around a few key areas. For the author, the process is reminiscent of moving from procedural to object-oriented programming models. The transition was confusing at first since new and old models share many of the same characteristics. However, there were just enough differences to throw off familiar patterns.

As educators, it is important to empathize with an audience making this change. Not all people find it exciting to learn a new technology particularly if there uncertainty involved or external pressures that restrict one's ability absorb the information. Identifying key concepts, drawing upon what is already known, and constructing frameworks leaves a trail for those who follow. IPv6 is no different than any other new technology introduced for next generation systems. It only happens to be replacing a technology that is widely regarded and familiar.

This paper draws on the author's experience in this transition and highlights the notable issues that differentiate IPv6 from IPv4. A variety of topics were presented that focus on patterns of thinking, deployment schemes, and technology discrepancies that should be watched for. Examples were provided and suggestions offered to help guide those familiar with IPv4 technologies though the transition period. At some point, IPv6 will be common place and practitioners will have mastered the essential concepts that make it functional. Until then, there is work to do.

References

[1] IETF, "RFC 4862 IPv6 Stateless Address Autoconfiguration," September 2007. [Online]. Available: https://tools.ietf.org/html/rfc4862.

[2] IETF, "RFC 4291 IP Version 6 Addressing Architecture," February 2006. [Online]. Available: https://tools.ietf.org/html/rfc4291.

[3] Wikipedia, "IPv6 address," March 2016. [Online]. Available: https://en.wikipedia.org/wiki/IPv6_address.

[4] IETF, "RFC 4941 Privacy Extensions for Stateless Address Autoconfiguration in IPv6," September 2007. [Online]. Available: http://tools.ietf.org/html/rfc4941.

[5] IETF, "RFC 3022 Traditional IP Network Address Translator (Traditional NAT)," January 2001. [Online]. Available: https://tools.ietf.org/html/rfc3022.

[6] Wikipedia, "Comparison of IPv6 Support in Operating Systems," February 2016. [Online]. Available: https://en.wikipedia.org/wiki/Comparison_of_IPv6_support_in_operating_systems.

[7] IETF, "DHCPv6/SLAAC Interaction Problems on Address and DNS Configuration," 2016. [Online]. Available: https://tools.ietf.org/html/draft-ietf-v6ops-dhcpv6-slaac-problem-06.

[8] IETF, "RFC 4294 IPv6 Node Requirements," 2006. [Online]. Available: http://tools.ietf.org/html/rfc4294.

[9] IETF, "RFC 3315 Dynamic Host Configuration Protocol for IPv6 (DHCPv6)," July 2003. [Online]. Available: https://tools.ietf.org/html/rfc3315.