The Inevitability of IPv6 PART 1

BY JOHN HOWIE

A switch from IPv4 to IPv6 is on your horizon. Are you ready for it?

Internet Protocol version 6 (IPv6) is the set of protocols that will replace today's IPv4. IPv6 offers many benefits necessary to support the Internet's continuing expansion—most notably an expanded address space that overcomes pressures in regions such as Africa, Asia, China, and the Middle East. Temporary solutions such as Network Address Translation (NAT)—although effective in the short term—won't provide long-term help. Recognizing that IPv6 is the future, many governments are mandating that their systems and networks support IPv6, including the US government, which has set a transition date of June 30, 2008. If your company does business with entities that use (or plan to use) IPv6, you'll feel the pressure to support IPv6, if only to support communications between your company and your partners. Simply put, IPv6 might become a competitive advantage.

FUN FACTS

- There are enough available IPv6 addresses to give every star in the known universe almost 7×10^{15} addresses.
- IPv6 was once called IPng, for Internet Protocol next generation.
- The successor to IPv4 couldn't be called IPv5 because the protocol version 5 was allocated to the Internet Stream Protocol in the 1970s. IPv6 reflects that the protocol is version 6.
- IPv4 uses 32 bits for addresses, whereas IPv6 uses 128 bits. There aren't enough available IPv4 addresses for everyone on Earth, but with IPv6, every person could have almost 5×10^{28} addresses each!

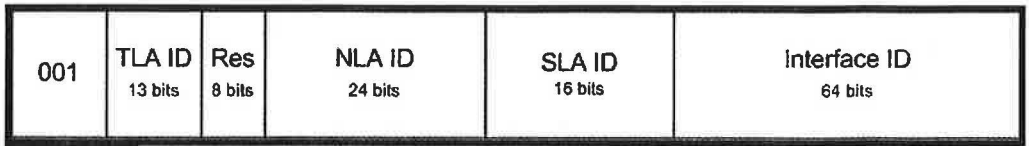

Figure 1: Global Unicast Addressing

In this first part of a three-part series, I describe IPv6 addressing in detail, focusing on how its addressing scheme works. I also describe some of the new features of IPv6, as well as some of the reasons you should care about it—even if you don't plan on implementing it in the near future. In two future articles, I'll describe how to install IPv6 onto Windows Server 2003 and Windows XP, and how to configure interfaces with addresses and enable DNS resolution. I'll also describe in detail how to configure your systems and networks to use IPv6 and IPv4 together while you transition to an all-IPv6 network. Finally, I'll look into strategies for using IPv6 over the IPv4 Internet if your ISP doesn't support IPv6. But first, we need to lay down a foundation.

Windows Support for IPv6

Almost every modern OS supports IPv6 out of the box. In fact, you're probably running IPv6 on your networks without even realizing it. Microsoft supports IPv6 in Windows Vista, Windows 2003, XP SP1 and later, and Windows CE .NET 4.1 and later. Windows Server 2008 will also support IPv6. Microsoft Research produced an IPv6 stack for Windows 2000 and Windows NT, but it isn't supported. To obtain the stack, see the Learning Path online.

Only Vista has IPv6 enabled "out of the box." If you have Vista installed on your network, you're running IPv6. Vista will configure link-local addresses in the absence of IPv6 infrastructure hardware such as DHCP servers, IPv6-capable routers, and so on. Once enabled, XP will function as an IPv6 client, letting you conduct many common communications (e.g., Web browsing using HTTP or HTTPS) over IPv6. Windows 2003 also supports IPv6 in most communications.

IPv6 Addressing

IPv6 gives you a whole new means of uniquely addressing a node (or end system). In IPv6, there are 128 bits available to uniquely identify a node. IPv4 offers 32 bits, for a total of more than 4 billion possible combinations, but far fewer are practically available because of the way address space has been organized.With 128 bits, we'll have sufficient addresses for the next millennium—even given the way addresses are allocated.

Before I discuss the allocation and use of IPv6 addresses, it's helpful to understand the format that's used to represent them. Whereas IPv4 uses a dotted-decimal system (e.g., 192.168.16.10), IPv6 uses a different format. An IPv6 address is split into eight 16-bit blocks: Each block is represented by four hexadecimal digits, and each block is separated by a colon (:)—for example, 2001:0000:0000:e388:0092:fb7f:a827:fad6. Within each block, leading zeroes can be omitted so that the address can be read as 2001:0:0:e388:92:fb7f:a827:fad6. Also, blocks of zeroes can be omitted, so that the address can be further simplified as 2001::e388:92:fb7f:a827:fad6. Note the use of the double colon to represent the blocks of zeroes. If you have more than one block of consecutive zeroes in an address, only one block can be omitted. (Otherwise, it would be impossible to reconstruct the original address.)

Currently, three types of IPv6 addresses can be allocated to a node: *unicast, multicast,* and *anycast.* A unicast address uniquely identifies a single interface (or network connection) on a node (or a virtual interface on clustered systems). A multicast address is similar to an IPv4 multicast address and can be shared by several interfaces on several nodes. A packet with a multicast destination address is delivered to all interfaces on all nodes that share the address. However, a packet with an anycast destination address is delivered to only one interface: the nearest interface to the sending interface. Regardless of type, the address identifies an interface on a node—not the node itself. A node will likely have multiple IPv6 addresses, even if it has only one interface.

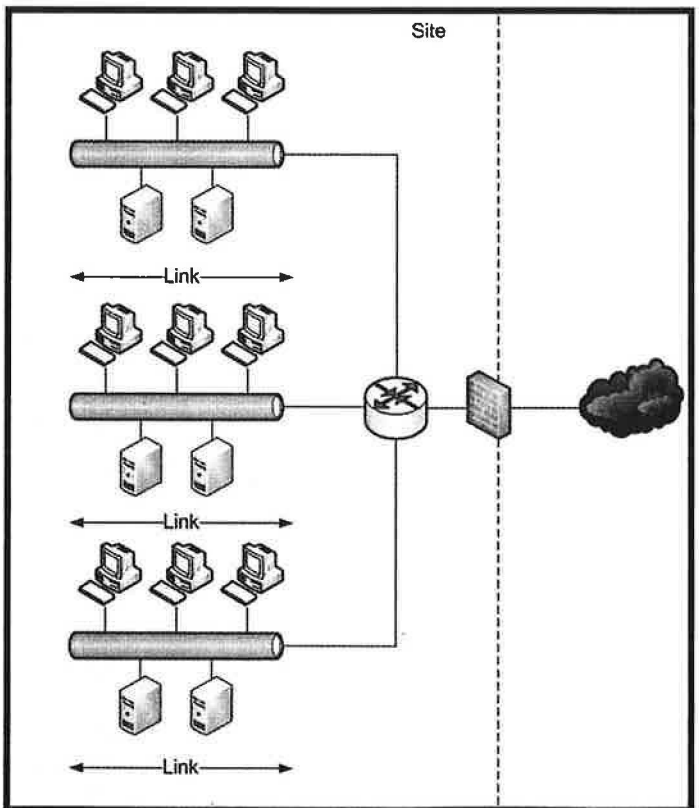

Figure 2: IPv6 Address Scope

Unicast Addresses

Each interface can have more than one unicast address. A unicast address can be an Aggregatable Global Unicast Address (aka global address), or a Local-Use Unicast Address.

Global address. A global address is unique to the interface it's assigned to and can be used to reach that interface from any other interface. Global IPv6 addresses are hierarchical and contain routing information. Figure 1 shows the format of a global address. A unicast address's first three bits—called the Format Prefix (FP)—are always 001. FPs can be of varying length (e.g., the multicast FP is eight bits in length). The next thirteen bits comprise the Top-Level Aggregation Identifier (TLA ID). This ID is allocated to top-level ISPs, of which there can be 8,192.

Next in the address is a reserved field—eight bits in length and designed for future expansion of the TLA ID. The next field in the address, the Next-Level Aggregation Identifier (NLA ID), is 24 bits in length and is used by the top-level ISP to organize networks or to support second-tier ISPs, each of which would have one or more NLA IDs assigned to them.

These combined 48 bits uniquely identify a site belonging to the top-level or second-tier ISP's customer. Sites are determined by geography. For example, an international company might have many sites. Each site's IPv6 connection will have a 48-bit address unique to the site. Each site can use the next sixteen bits in the address—called the Site-Level Aggregation Identifier (SLA ID)—to divide the site into subnets. Each site can have 65,535 subnets. Alternatively, if a company has multiple sites but only one IPv6 connection via an ISP, it can use the SLA ID to route between the sites and to the connection. The last field in the global address is the Interface ID, which is 64 bits in length. This field is similar to IPv4's host identifier, which uniquely identifies the host on the network.

Local-Use Unicast Address. There are two types of Local-Use Unicast Addresses. The first is called a *link-local address*, which is used to communicate between interfaces belonging to nodes on a single link. The second is called a *site-local address*, which is used to communicate between interfaces belonging to nodes in a site. Both are viable alternatives to a global address, depending on the scope. Figure 2 shows the scope of a link and a site.

IPv6 gives us sufficient addresses for the next millennium.

Link-local addressing is similar to IPv4's Automatic Private IP Addressing (APIPA). Link-local addresses begin with an FP of FE80:—the last 64 bits of a link-local address are the Interface ID, and the bits in between the FP and the Interface ID are zeroed out. As with APIPA, link-local addresses are automatically configured without the need for a DHCP server or manual configuration. In fact, every IPv6-capable interface automatically has a link-local address configured for it. If you have any nodes on your network that support interfaces with IPv6, they'll have a link-local address and might be sending packets onto your network as part of Neighbor Discovery. Two nodes on the same link with interfaces that support IPv6 will automatically be able to communicate with each other, without any further configuration or management. However, communication using link-local addresses is restricted to a link—IPv6-aware routers should never forward packets with link-local source or destination addresses.

Site-local addresses are similar to the IPv4 private addresses, which have the network identifiers 10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16. Site-local addresses always begin with an FP of FEC0:. As with link-local addresses, the last 64 bits of the address com-

prise an Interface ID. The lower 16 bits of the top 64 bits—called the Subnet ID field—uniquely identify subnets in the site, the same as the SLA ID field in a global address. The bits between the FP and the Subnet ID field are zeroed out.

IPv6 uses two special constant addresses. The first is called the *unspecified address* and is always set to 0:0:0:0:0:0:0:0, or just :: for short. This address—similar to the IPv4 address 0.0.0.0—functions as a source address when no other address is available (e.g., when requesting an IP address from an IPv6-capable DHCP server). The second address is the *loopback address* and is always 0:0:0:0:0:0:0:1, or simply ::1. This address—equivalent to the IPv4 loopback address 127.0.0.1—can be used for local testing of applications and configuration. Every interface will respond to the loopback address.

The Interface ID

The Interface ID in a unicast address is always 64 bits in length. It was designed this way to support 48-bit MAC addresses of current 802.x LAN technologies such as Ethernet, and wireless technologies such as Bluetooth and Wi-Fi, as well as the 64-bit addresses that FireWire uses. Future 802.x series LAN and wireless technologies will also use 64-bit addressing. The requirement to support 48-bit and 64-bit MAC addresses comes from the requirement that the Interface ID in a unicast address can be derived from a MAC address using an Extended Unique Identifier (EUI) 64 address. The Interface ID can also be assigned manually or by an IPv6-capable DHCP server.

In the most common scenario, the Interface ID is derived from the 48-bit MAC address of an Ethernet card. A 48-bit MAC address is split into two 24-bit halves. The IEEE assigns the first 24 bits to manufacturers. The manufacturer uses the second 24 bits to uniquely identify the card. Although it's possible to override the MAC address of an Ethernet card, let's assume that it hasn't been overridden. To convert a 48-bit MAC address to a 64-bit Interface ID, the system first copies 24 bits of the MAC address to the first 24 bits of the Interface ID. Bits 17 and 16 of the first 24 bits representing the manufacturer (reading from right to left, starting at 0) are always set to 00. During the copy, the system sets them to 10. After the 24 bits are copied over, 16 bytes are added, and they're always 0xFFFE. The system then copies 24 bits in the second half of the MAC address to produce the 64-bit Interface ID.

In dial-up scenarios, the Interface ID can be generated using a process designed to guarantee the anonymity of the user. If not for this provision, a system could be tracked as it used the Internet, regardless of the ISP used, because the Interface ID would be unique to the computer regardless of the ISP.

Multicast Addresses

IPv6 multicasting is similar to IPv4 multicasting. A node that wants to listen for multicast traffic will set the IPv6 address of an interface to the multicast address that the traffic is being sent to. Multicast addresses have an FP of 0xFF. The next four bits of the multicast address comprise the Flags field. The lowest bit in the Flags field is called the Transient flag. If set to 0, the multicast address is a well-known address set by IANA; if set to 1, it's a non-permanent or transient multicast address. The next four bits of the multicast address comprise the Scope field. The purpose of this field is to identify the scope of the multicast traffic, and to identify the traffic as node-local, link-local, site-local, organization-local, or global. Routers use this field to determine whether to forward traffic. The last field in the multicast address is the Group ID, which is 112 bits in length. The Group ID identifies the multicast group. As with unicast addresses, there are predefined multicast addresses. Table 1 lists the three most common ones.

Table 1: Common Predefined Multicast Addresses

Multicast Address	Use
FF01::1	Node-local scope for all nodes
FF02::1	Link-local scope for all nodes
FF05::1	Site-local scope for all nodes

There's more to IPv6 than simply an expanded address space.

When using multicasting in IPv6, you should use only the bottom 32 bits of the Group ID field and zero out the top 80 bits. Doing so eases conversion support of the multicast address to an Ethernet multicast address. An Ethernet multicast address takes the form 33:33:xx:xx:xx:xx. Using the recommended multicast addressing format, the bottom 32 bits of the Group ID create the Ethernet multicast address.

IPv6 also uses multicast addresses to support link address resolution. Every interface adds a multicast address for each of its unicast addresses. The multicast address takes the form FF02::1:FFxx:xxxx. The system copies the last 24 bits of the unicast address to the multicast address to replace the xx:xxxx. The system then maps the IPv6 multicast address to the MAC multicast address, as described above. This scheme reduces the number of nodes that have to process address-resolution requests. In IPv4, when one node wants to obtain another node's interface MAC address, the system sends a broadcast message to the broadcast MAC address. Therefore, every interface on the link is forced to process the request—even if it's not intended for it. In IPv6, a node that wants to find another node's interface MAC address will send a broadcast message to the multicast address FF02::1:FF:xx:xxxx, where xx:xxxx is the bottom 24 bits of the interface ID. This, in turn, is translated into a MAC multicast address 33:33:FF:xx:xx:xx. Only those interfaces on the link with matching lower 24 bits in their Interface ID need to respond to the address-resolution request.

IPv6 Features

There's more to IPv6 than simply an expanded address space. IPv6 includes a new header format, improved support for extensions and options, flow-labeling capabilities, and authentication and privacy capabilities.

New header format. IPv6's new header format minimizes the overhead often spent processing fields or information in packet headers. In IPv4, routers and end systems are required to examine packets in detail, looking for information necessary to determine whether the packet should be processed further. With IPv6, you'll now find those fields (when required) after the main packet header in Extension Headers. The new header format makes header processing much more efficient at routers, which can

ignore information in any Extension Headers—with the exception of a Hop-by-Hop Extension Header, which must immediately follow the IPv6 header. The Hop-by-Hop Extension Header might contain information necessary for a router, such as a warning that a packet is a Jumbo packet (greater than 65,535 bytes), or that a router must perform additional processing on the packet.

Improved support for extensions and options. The change in the IPv6 packet header format and the use of Extension Headers facilitate this new feature. Options in Extension Headers have fewer limitations on size than in IPv4, and IPv6 is extensible by adding more defined Extension Headers over time. In IPv6, if a destination node receives an IPv6 packet containing an Extension Header that it doesn't recognize, it informs the source node via Internet Control Message Protocol version 6 (ICMPv6) that it can't process the packet. This feature lets nodes implement IPv6 extensions independently of each other and still communicate.

Flow-labeling capabilities. IPv6 uses flow labeling for Quality of Service (QoS). Flow labeling lets a source node define a priority (e.g., real time), which might be used in Voice over IP (VoIP) or video-over-IP solutions to guarantee delivery of a packet within a certain time window. In IPv4, QoS often requires a router or node to look beyond a packet's header for information. In IPv6, all necessary information is in the header.

Authentication and privacy. IPv6's authentication and privacy capabilities are, essentially, IPSec. IPSec is now a requirement in IPv6 implementations, whereas in IPv4 it's an optional component. IPSec supports Authenticated Headers, which authenticate nodes to each other and ensure the integrity of data exchanged between them, and Encapsulating Security Payload (ESP), which has similar functionality but also includes the ability to encrypt data for confidentiality.

Unlike IPv4, in which different implementations of the protocol by different vendors could—and would—result in an inability of nodes to communicate with each other, in IPv6 interoperability is almost guaranteed, thanks to the underlying standards.

Stay Tuned

We've only just started. Now that you've got some solid foundational knowledge about IPv6, you're primed to dive into the actual installation and use of the protocol. Get ready to make it work on Windows 2003 and XP, and prepare yourself for configuring interfaces with addresses and enabling DNS resolution. In a later article, I'll also describe talk about enabling IPv6 and IPv4 interoperability on your way to an all-IPv6 network.

InstantDoc ID 96880

John Howie

(jhowie@microsoft.com) is the director of the World Wide Services and IT Technical Community for Security at Microsoft. He has more than 15 years of experience in information security and is a CISA, a CISM, and a CISSP.

The Inevitability of IPv6 PART 2

BY JOHN HOWIE

Configure IPv6 in your network—even if your routing infrastructure doesn't yet support it

As I maintained in "The Inevitability of IPv6, Part 1" (InstantDoc ID 96880), even if you have no immediate plans to migrate to IPv6 in your enterprise, you need to be ready for it, and you need to understand how Windows uses it. Some current and forthcoming Microsoft products and updates to existing products do or will ship with IPv6 enabled and running out of the box. If you communicate regularly with business partners over the Internet, you might be forced to tackle IPv6 because many companies are already beginning to make the transition. Increasingly, governments—including the US Government—are mandating its use.

In Part 1, I described how Microsoft is supporting IPv6 in its product line, and I provided an overview of how IPv6 addressing works. Be sure you're well-versed in that article's foundational information before taking the plunge into this month's discussion, which is Part 2 of a three-part series. Now, without further ado, let's investigate how to install and configure IPv6 in your Windows network and how to use IPv6 to communicate—even if your routing infrastructure doesn't yet support it.

FUN FACTS

- By June 30, 2008, every US government agency network backbone must be able to handle IPv6 traffic.
- A number of IPv6 task forces are in operation around the world. You can find details about many of them at www.ipv6tf.org.
- Most higher academic institutions in the western hemisphere and much of Asia have been using IPv6 since 2003.

Installing IPv6 on Windows 2003 and XP

As I explained in Part 1, Windows Vista comes with IPv6 installed and running, as will Windows Server 2008 when it ships. However, if you're running Windows Server 2003 and Windows XP, you'll need to manually install and configure IPv6. Let's get to it!

To install the IPv6 protocol on the earlier OSs, select the adapter on which you want to use IPv6, open its Properties dialog box, and click Install to open the Select Network Component Type dialog box. You can install Client, Server, and Protocol components. Select Protocol and click Add to open the Select Network Protocol dialog box, select *Microsoft TCP/IP version 6* from the options, and click OK. Figure 1, page 56, shows the dialog boxes on a Windows 2003 system. You don't need to visit each machine to install IPv6. You can simply run the Netsh Interface IPv6 Install command from a startup script or within a package that you can distribute to each

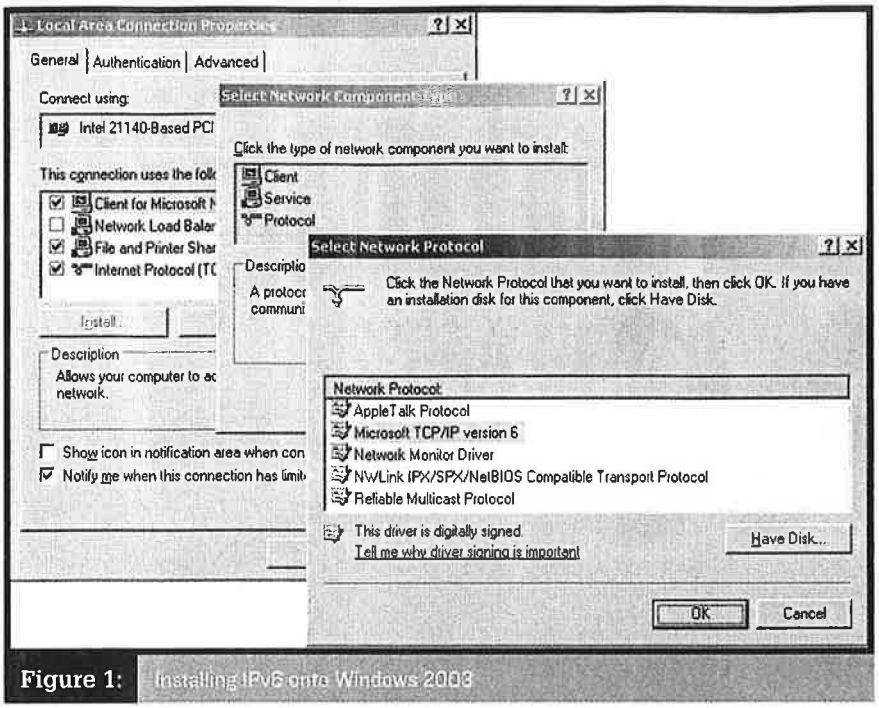

Figure 1: Installing IPv6 onto Windows 2003

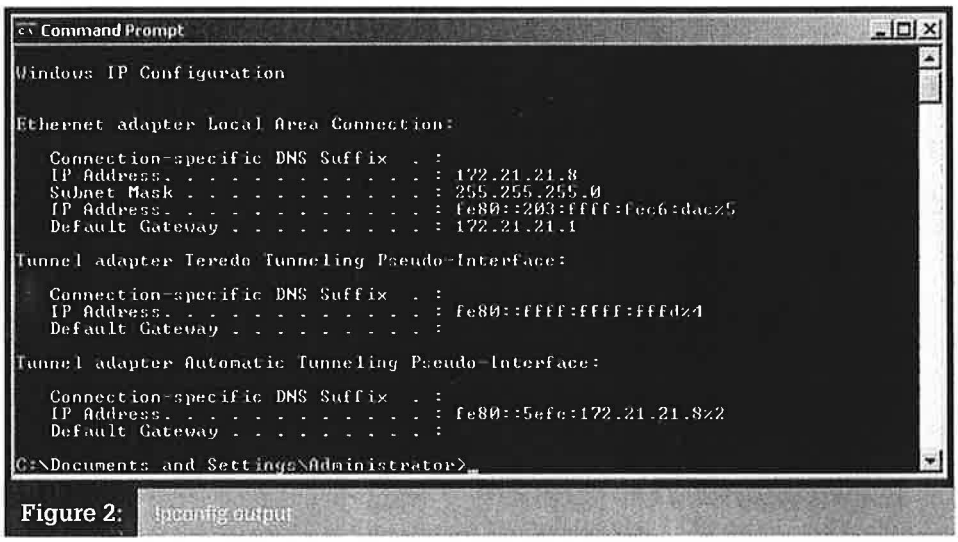

Figure 2: Ipconfig output

system—perhaps via Microsoft Systems Management Server (SMS) or Group Policy.

Once you've installed IPv6, you'll notice that—unlike IPv4—you can't configure the protocol's properties from the network connection's Properties dialog box. With IPv4, you would configure an interface with a static IP address or configure it to use DHCP to get an address from a DHCP server. With IPv6, that's not the case. As I described in Part 1, an IPv6 unicast link-local address can be derived from the NIC's 48-bit MAC address. Windows 2003 and XP automatically use this process to assign a link-local IPv6 address to an interface. This address, which begins with FE80::/64, can be used to communicate with every other host running IPv6 on the same link but can't be used to communicate with hosts through a router. IPv6 routers never route traffic with unicast link-local addresses.

A word of caution here: If you use Microsoft Virtual Server or another virtualization suite to host virtual machines (VMs), you need to make sure that each VM has a unique MAC address for each virtual NIC. If you don't, each node running IPv6 will have the same IPv6 address—and that will cause you problems.

You can determine a node's unicast link-local address by running Ipconfig from the command line. Figure 2 shows the Ipconfig output on a Windows 2003 system. You'll see three unicast link-local addresses; ignore the addresses for the Automatic Tunneling and Teredo adapters. (I'll cover these later.) The %*n* component (where *n* is a number) at the end of each IPv6 address refers to the NIC or adapter to which the address is assigned. Windows uses numbers to identify both physical and virtual network interfaces or adapters. The adapter number is important for reasons that I will discuss shortly.

If you run Getmac from the command line, you'll see how the MAC address for each NIC has been used to build the IPv6 unicast link-local address. You can verify that IPv6 is working on a node by using the Ping command to test the IPv6 loopback address, which is ::1. You can use Ping to verify communication with other IPv6 nodes if you know their IPv6 unicast link-local addresses. However, there's a trick to successfully using Ping. When you specify the IPv6 address of the node for which you want to test connectivity, you must append your node's adapter number. For example, in Figure 2, the unicast link-local address of the Local Area Network adapter has the suffix %5, meaning that the address is bound to adapter number 5. To ping the node with the unicast link-local address fe80::203:ffff:feab:3045, you would type in the command

```
ping fe80::203:feab:3045%5
```

As quick and easy as it is to get IPv6 up and running with unicast link-local addresses, an enterprise still has to take several steps to build a useful IPv6 network. Unicast link-local addresses won't work in routed networks and can't be used over the Internet. Routable addresses need to be assigned to hosts, routers need to be configured, and DNS needs to be configured to enable mapping of FQDNs to IPv6 addresses.

Configuring Windows with Routable IPv6 Addresses

Configuring Windows with routable IPv6 addresses can be easy or difficult, depending on your enterprise's circumstances. Typically, you configure a host to obtain an IPv4 address from a DHCP server, or set a static address. As I mentioned earlier, there's no means to configure IPv6 from a network connection's Properties dialog box in Windows 2003 or XP. (Vista and the forthcoming Server 2008 do let you set the IPv6 address in this fashion.)

Additionally, there's currently limited support for using DHCP with IPv6. There's no supported IPv6-capable DHCP server available for Windows 2003 from Microsoft, and you'll have to wait for Server 2008 for an IPv6-capable DHCP server.

As I discussed in Part 1, a routable IPv6 address can be a unicast site-local or globally aggregatable address. A site-local address is similar to private IPv4 addresses that fall in the range 10.0.0.0 to 10.255.255.255, 172.16.0.0 to 172.31.255.255, and 192.168.0.0 to 192.168.255.255, and begin with FEC0::/48. Each site-local address has a 16-bit subnetwork identifier, followed by a 64-bit interface identifier. A site can have as many as 65,535 subnetworks. The interface identifier is derived from the NIC's MAC address, just like a link-local address. Like the IPv4 equivalent, site-local addresses can't be used to communicate over the Internet; thus, site-local addresses are useful if you want to use IPv6 internally. Without an IPv6-capabale DHCP server, configuring a site-local address for a node is accomplished either as a manual, per-machine process using Netsh or, more usually, through address auto-configuration.

Figure 3: Ipconfig output on Vista with site-local address configured

Auto-configuration is a means by which nodes can determine both site-local and globally aggregatable addresses from IPv6 routers on the same link—that is, routers that can be pinged using their link-local address from a node. When a node starts or when the network interface is reset (e.g., a previously disconnected laptop connects to a corporate network), the node will send out a router-solicitation message on each network interface—in essence, a request for routers on the same link to make themselves known. Properly configured IPv6 routers will respond with a router advertisement address, consisting of a site-local prefix address in the form of fec0:0000:0000:*subnetid*::/64. The node will configure each interface on which it receives a router advertisement in response to a router-solicitation message with an IPv6 address consisting of the site-local prefix received from the router and the 48-bit MAC address of the interface the solicitation was sent over and the advertisement received.

Figure 3 shows the output of the Ipconfig command on Vista, clearly identifying the site-local address assigned to the Local Area Connection adapter. Neither Windows 2003 nor XP differentiates between link-local and site-local addresses and instead just displays them as *IP Address*. Windows will configure the IPv6 router to be the default gateway for IPv6 traffic. Assuming your routers have accurate routing IPv6 tables, you don't need to configure your Windows nodes further.

To verify that you can connect with IPv6 nodes not on the local link, you can use Ping and specify site-local addresses instead of link-local addresses. You don't need to append the IPv6 address of the destination node with the adapter number of an interface when using site-local addresses, because each address should be unique across all your subnets—assuming your routers are configured correctly with subnetwork identifiers.

Connectivity with ISATAP

If your network doesn't have IPv6-capable routers, it's still possible to use IPv6 in your enterprise to facilitate communication between nodes on different links, by using a technology called Intra-Site Automatic Tunnel Addressing Protocol (ISATAP), which is configured automatically for you. You can use ISATAP only on nodes running both IPv4 and IPv6. If you look back at Figure 2, you'll see the entry for the Automatic Tunneling adapter. This is the ISATAP-configured IPv6 address. (You can identify it in Vista by running the command Ipconfig /all and looking for a tunnel adapter whose description begins with the name *isatap*.)

Every Windows OS with IPv6 enabled and an IPv4 address will automatically create an ISATAP address. ISATAP addresses take the form ::5EFE:*w.x.y.z*, where *w.x.y.z* is the host's IPv4 address. ISATAP addresses can have any prefix, including a link-local address, site-local address, or a globally aggregatable address, although a prefix that is a site-local or globally aggregatable address implies that IPv6 routing is possible, and ISATAP isn't necessary. For this reason, ISATAP addresses on Windows use the link-local prefix fe80::.

One IPv6 node can reach another by using its ISATAP address, even if it's on a different link and there's no IPv6-capable router connecting the links. ISATAP is a great transition technology, letting you use IPv6 on IPv4 networks when there's no support at the router or gateway level. In Part 3 of this series, I'll further discuss ISATAP and show you how to use it to communicate over the Internet.

DNS and IPv6

With site-local IPv6 addresses allocated to your nodes, each can use IPv6 instead of IPv4 to communicate with the other. Member servers running Windows 2003 R2 and XP and Vista clients will register their site-local IPv6 addresses in DNS, if possible. IPv6 addresses are stored as AAAA records. (IPv4 addresses are stored as A records.) DCs, even if they're IPv6 nodes, don't register AAAA records for themselves in DNS. You can always use the DNS Microsoft Management Console (MMC) snap-in to manually add AAAA records for IPv6 nodes, including non-Windows hosts. Note that if you replicate your DNS zones containing AAAA records to non-Windows 2003 servers, you might encounter difficulties because not all DNS servers support AAAA records.

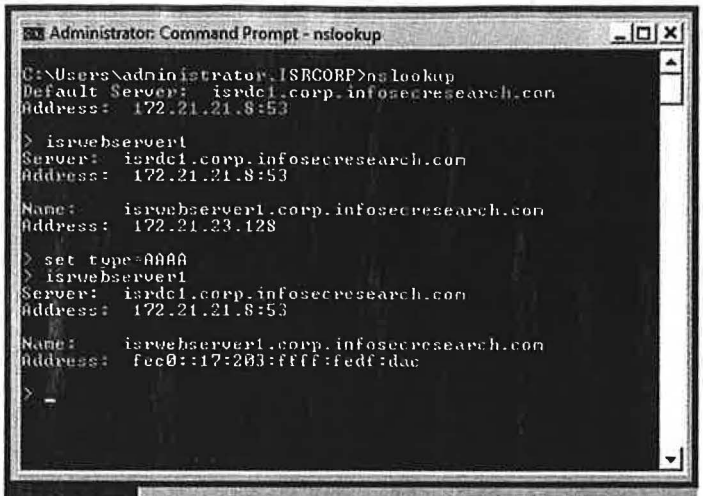

Figure 4: Using Nslookup to query a host's IPv4/IPv6 addresses

You can use the Ping command to test lookups of IPv6 addresses for FQDNs; simply specify the FQDN of the target host on the command line. The address retrieved and subsequently used in the ICMP echo request should be an IPv6 address. You can also use the command-line tool Nslookup to query DNS for AAAA records by simply typing *set type=AAAA* into Nslookup before typing the name of the server for which you're querying. Figure 4 shows the use of Nslookup to query for a Web server's IPv4 address (the default), followed by a request for the IPv6 address.

Although an IPv6 address might be available for a host in DNS, there's no guarantee that the network services running on it will support IPv6. For this reason, I don't recommend that you create AAAA records for your Windows 2003–based DCs; many of the services that run on a DC don't yet support IPv6.

In the Thick of It

So, now you know how to install IPv6 on your DCs, member servers, and clients running Windows 2003 R2 and XP SP2. You also know how to test link-local connectivity between IPv6 nodes, and how nodes use auto-configuration in communication with IPv6-capable routers to automatically assign site-local IPv6 addresses.

In Part 3, I'll describe how to configure globally aggregatable addresses so that your nodes can communicate over the Internet with other IPv6 nodes, as well as a means to facilitate interoperability with IPv4 nodes and to run IPv6 over IPv4 when your ISP doesn't support IPv6. Stay tuned!

InstantDoc ID 97365

John Howie
(jhowie@microsoft.com) is the director of the World Wide Services and IT Technical Community for Security at Microsoft. He has more than 15 years of experience in information security and is a CISA, a CISM, and a CISSP.