December 28, 2009 12:00 AM

Hands-On IPv6 Lab Setup

Build your own IPv6 lab and start preparing for the big move now

Mel Beckman

Windows IT Pro

InstantDoc ID #103361

Rating: (1)

At this writing, Internet experts predict that we have less than two years before the exhaustion of IPv4 addresses that fuel the growth of the Internet. Of the 4,294,967,296 addresses available in IPv4's 32-bit address space, we've consumed 90 percent, leaving less than 425 million addresses remaining. That's not a lot, and the rate of consumption is increasing, making it difficult to pin down the actual date the last address will be used.

Alas, trouble will start long before that day arrives, owing to the economics of scarcity and demand. As a resource becomes less plentiful, its price increases, something that has already occurred with IP addresses in North America. Unless an alternative resource—IPv6 addresses in this case—becomes available, the cost of getting a new public IPv4 address could skyrocket.

Fortunately, the IPv6 Internet is alive and well, and waiting for your arrival. We all had a golden opportunity to push an IPv6 migration more than a year ago, before the current economic crash, but few availed themselves of that opportunity. Now, with funds tight and jobs precarious, we're faced with making the IPv6 move on shoestring budgets. Consider this article your shoestring to IPv6. By spending very little money and a modicum of your own time, you can set up an IPv6 lab that will help position you for an IPv6 transition when it becomes unavoidable.

With an IPv6 laboratory at your disposal, you'll gain experience with IPv6 addressing, network troubleshooting, and deployment methods. You'll also have a test bed for validating IPv6 compatibility for current and future applications, hardware, and services. Most importantly, you'll be augmenting your marketable skill set in a way that will make you much more valuable to your employer, current or future.

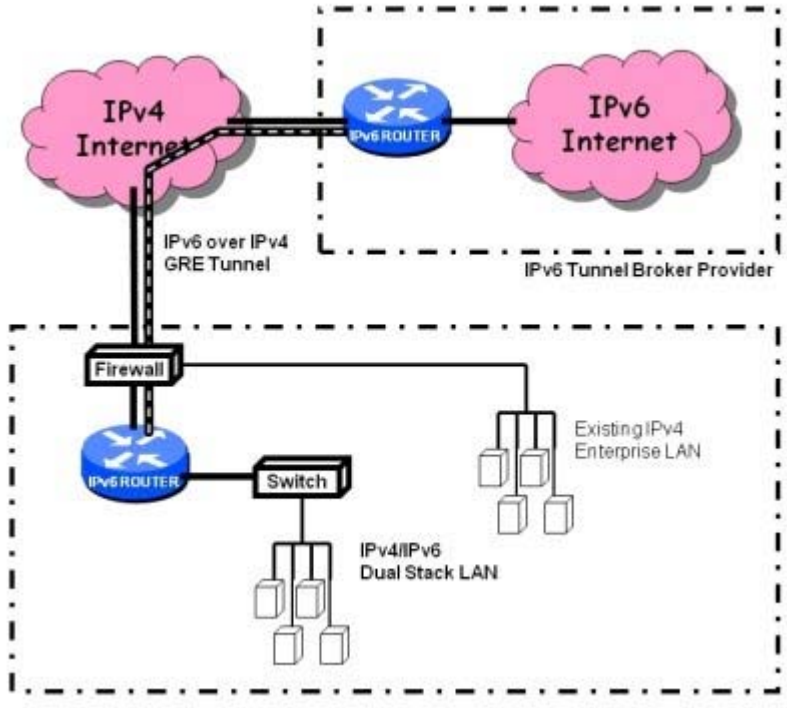
The Big Picture

In "The Inevitability of IPv6 Part 1" (and Part 2), John Howie explained IPv6 addressing, features, and operation under various versions of Windows. If you're not conversant in IPv6 addressing concepts and terminology, you should review those articles as a prerequisite to starting this project.

In addition to that technical background, an essential thing to know is that the IPv6 Internet is open for business, populated by the likes of Google, Apple, and Microsoft, and growing at a steady pace. You can get connected to that Internet by a variety of methods: a direct connection from your ISP, a desktop-only connection using VPN tunneling, or a LAN connection via an IPv6 ISP offering tunnel-brokering services. The most useful approach for a lab is the last method, because you get a large IPv6 address space that you can use for actual deployment when you're ready and that you can readily break into subnets for experimental purposes.

The image below illustrates how your lab connects to an IPv6 tunnel broker. The tunnel broker operates one or more IPv6 tunnel servers located around the world. You choose a broker with a server reasonably close to you and sign up for service, which typically costs nothing—the tunnel broker hopes to win your business when you move your entire organization to IPv6. You then set up an IPv6 tunneling router behind your enterprise firewall, which establishes an IP protocol 47, or

Generic Routing Encapsulation (GRE), IPv6-over-IPv4 tunnel connection to the tunnel broker and delivers IPv6 connectivity to your IPv6 laboratory network. Protocol 47 is often used for VPN tunnels, but in this case, the tunnel isn't encrypted. It's just used to transport IPv6 packets to your lab network.



A point you should have picked up in your IPv6 prerequisite reading, but worth reiterating here, is that all IPv6 networks are actually a combination of both IPv4 and IPv6 protocols, a configuration termed dual stack. A dual-stack network is necessary because the opportunity to make a clean cutover to IPv6 passed us by years ago. The only transition path left available is for every device to be on both IPv4 and IPv6 networks for the several years it will take to move totally to IPv6. Although more complex than the clean-cut approach, dual stack operation has the advantage of letting you gradually introduce IPv6 into production networks.

It's important to recognize that you'll be routing public IPv6 addresses behind your corporate firewall, so your IPv6 lab network should remain completely isolated for security purposes. The example router configuration provided in this article includes a sample security policy that blocks all inbound traffic except HTTP (TCP port 80). You can modify that policy to suit your experimental requirements.

To make your IPv6 lab operational, you'll have to complete three tasks:

1. Sign up with an IPv6 tunnel broker and get an IPv6 IP address allocation.
2. Acquire and configure an IPv6 router to use the tunnel.
3. Activate and test your IPv6 connection.

Signing Up for Service

You have several choices for IPv6 tunnel brokering service. It's possible that your current ISP offers this service, and if so, that may be your best bet, because you'll get a connection that performs well and that you can easily transition to production use in the future. Alas, only a few ISPs have this option available. Fortunately, tunnel brokers are willing to give the service away to new IPv6 adopters like you. A partial a list of these providers and their contact information is shown here.

Provider	Coverage	Site
AARNet	Australia	broker.aarnet.net.au
Freenet6	Canada, Netherlands, Indonesia	gogonet.gogo6.com
Hurricane Electric	US, Canada, Europe, Asia	tunnelbroker.net
JANET	UK	www.broker.ipv6.ac.uk
Nerim	France	admin.nerim.net/nav/ipv6/

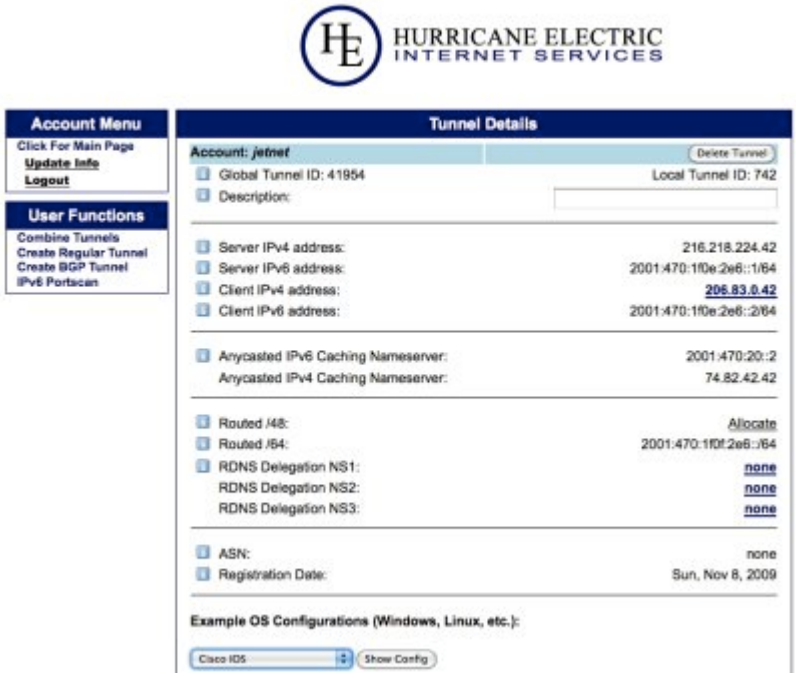
SixXS	US, Europe, New Zealand	sixxs.net

To sign up for service, give the tunnel broker your contact information and you'll receive an account user ID and password. You can then log into that account to create an IPv6 tunnel and get an IPv6 IP address allocation. In North America, the most prolific tunnel broker is Hurricane Electric (HE.net). Using their process as an example will guide you to requesting connectivity from any IPv6 tunnel broker.

Once you've logged into your newly created account, you'll be given an option to create a tunnel, as shown here. All you need provide is a static public IP address on your end. This can be a public IP address associated with your current firewall or a separate static IP address that you assign directly to the IPv6 router at your end—more on that option later. You can change this address in the future, so don't worry too much about choosing exactly the right address.

Click to expand.

After entering your public IP address and clicking Submit, your tunnel will be created and you'll get a Tunnel Details page listing IPv4 and IPv6 addresses that define your tunnel, as shown here. The important values to note from this are the following (you'll need these to configure your IPv6 router):



Click to expand.

- Server IPv4 address
- Server IPv6 address

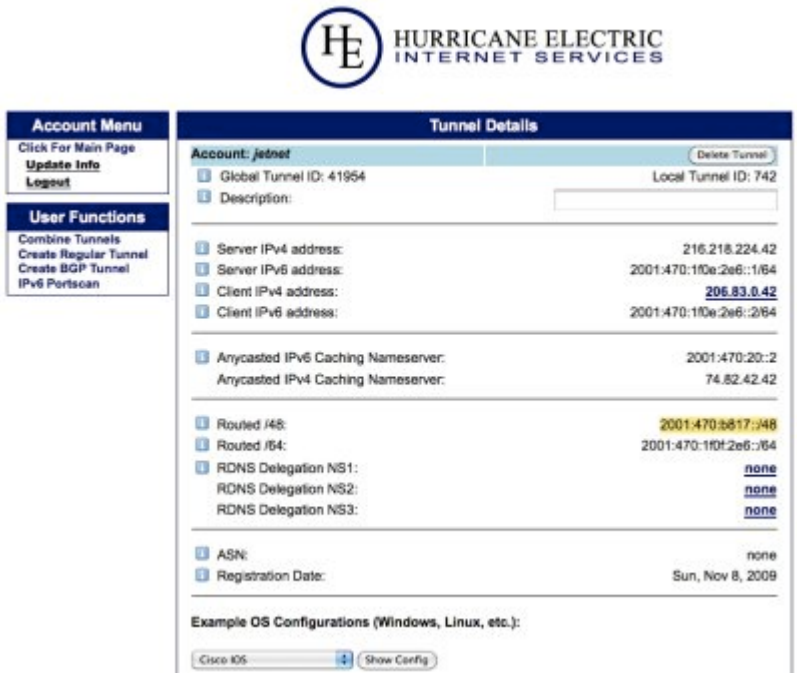
- Client IPv4 address
- Client IPv6 address
- IPv6 name server

With most tunnel brokers, you'll also automatically receive an initial IPv6 allocation. In this example, HE.net automatically allocates a 64-bit, or /64, IP subnet to you. That may seem like a large block, with twice as many bits as the total IPv4 address space of 32 bits, but it's actually small in IPv6 land. Even though it contains four billion times the number of IP addresses in the entire IPv4 Internet, a /64 is considered to be the size you'd allocate to a single LAN subnet. IPv6 is big. Really big.

It turns out that you don't want to use this initial /64 allocation, which is intended for very limited experimentation. What you want is a /48 subnet, which uses 48 bits for the network part of the address but gives you 80 address bits (128 minus 48) for devices. Now we're talking serious space. If you can follow the binary math here, given a 64-bit address space for each subnet, the 80 device address bits of a /48 allocation lets you have 32,768 /64 subnets. With this much space, you can readily slice and dice your IPv6 address space for whatever purposes you need.

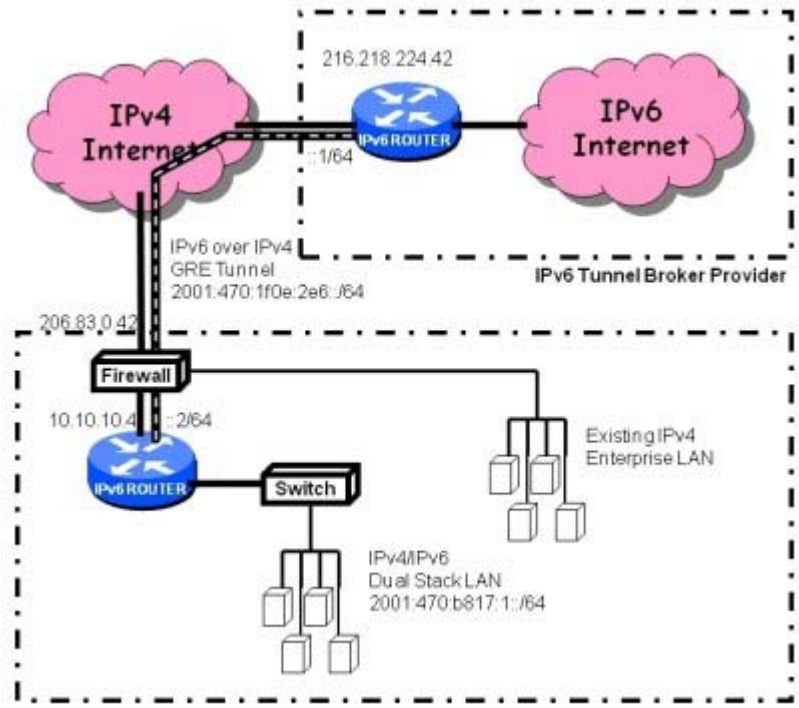
Notice that in the last image, the server and client IPv6 addresses are in the same /64 subnet, 2001:470:1f0e:2e6::/64, with the server being at ::1 and the client (your router) at ::2. No other IP addresses are used in this entire /64 subnet. That's right, you're just going to waste billions and billions of addresses on a single point-to-point link! This is standard practice in IPv6 land, and not a problem at all, given IPv6's vast capacity.

To actually get a /48 allocation for yourself, just click on the Allocate link in the "Routed /48" section of the Tunnel Details page. You'll get the same page back, updated with your fresh /48 allocation (2001:470:b817::/48, shown here). You're now ready for the first step that requires actual effort: getting and configuring an IPv6 router.



Click to expand.

Below is the Big Picture diagram with all the IPv4 and IPv6 addresses filled in, so you can see how traffic will flow in the live network. Note that the IPv6 network address of the experimental LAN, 2001:470:b817:1::/64, is actually just a subnet of the 2001:470:b817::/48 allocation. The fourth byte-pair of the address, :1:, represents the subnet number (0001 in this case), lengthening the /80 address to a full /64. Technically, you could use a subnet number starting at :0:, but I prefer to reserve subnet zero for utility purposes, such as network monitoring and infrastructure management.



The IPv6 Router

Although IPv6 may be new to you, it's not new to manufacturers of networking gear, who have collectively supported IPv6 for the last ten years or so. Devices from popular manufacturers are plentiful on the used market.

The specific router used here, the Cisco 1841 modular router, isn't the cheapest device out there (about $500 on eBay), but it has the best combination of software and hardware features for a lab device. A cheaper, but less capable, alternative is the Cisco 2621 (about $100), but this model lacks some IPv6 features, such as DHCPv6, which you might want to employ. Both routers have two Ethernet ports and modular slots for add-in cards, which you won't be using for your lab setup. The version of Cisco IOS software tested for this article is 12.4.24T1, which you should be able to obtain from any Cisco dealer, if it isn't already installed on the router.

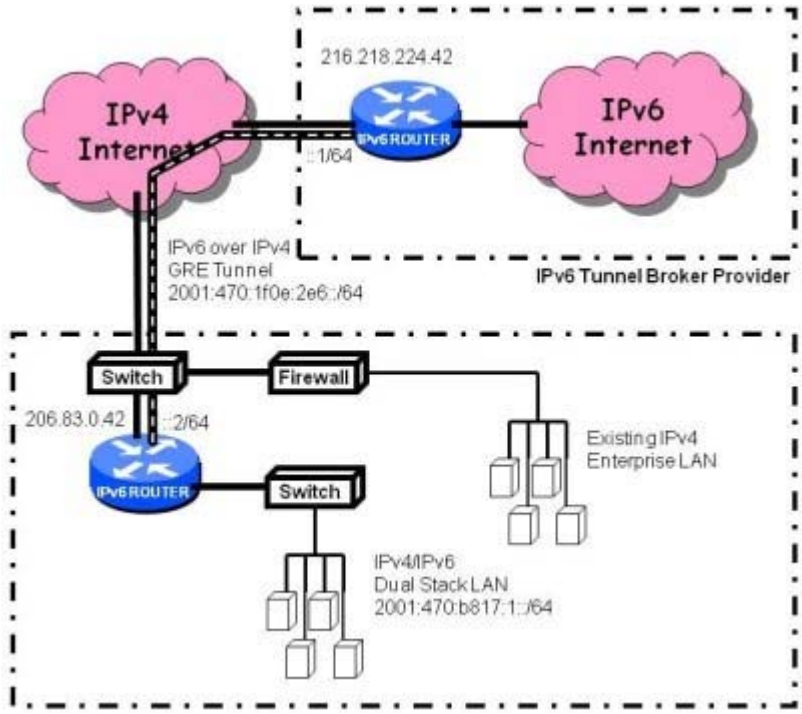
Once you've got the router, you'll need to configure it. This file shows a skeleton configuration you can modify to create a working configuration for your router; text in bold identifies the sections you'll change by plugging in the values previously collected from your tunnel IP address values. You'll also need to plug in your domain name for DHCPv6 to function correctly.

To enter your configuration into the router, connect the serial port of the router to a computer serial port (or USB serial adapter) and launch a serial access program such as Windows HyperTerm (supplied with Windows) or PuTTY (a free download from chiark.greenend.org.uk). If you're not familiar with configuring Cisco routers, a quick tutorial is online at cisco.com/warp/cpropub/45/tutorial.htm,

A configuration housekeeping detail to note is that the settings in your router shouldn't be the public settings. Instead, they should be a private, static IP address, mask, and gateway on your firewall-protected LAN. This is because your lab router is located behind your enterprise firewall, on the assumption that this is the easiest arrangement for most IT staffers. However, your firewall must support GRE (protocol 47) pass-through and be able to route GRE traffic to the specific static private LAN IP address of your IPv6 router. Most enterprise-class routers (such as Cisco ASA, Juniper Netscreen, and SonicWall TZ/NSA) have this capability. If yours doesn't, you can employ an alternate network topology (Figure 8), in which you connect your IPv6 router directly to a public DMZ Ethernet segment between your existing firewall and your ISP's internet access device. This usually requires installing a small Ethernet switch for this purpose, if you don't have one in place already. With this arrangement, the Client IPv4 address in your Cisco router configuration will be the public value from your tunnel details page.

Activate and Test

Once you've configured and connected your IPv6 router, it's time to take it for a test drive. Assuming you've either configured your enterprise firewall for GRE pass-through using the above instructions or located your IPv6 router directly on your public IPv4 DMZ (as shown below), your IPv6 tunnel should come online automatically and be ready for use. You can do the initial testing directly from the Cisco serial port command line interface or a telnet session logged into the router. First, check the status of the IPv6 tunnel (text you type is in bold).




```
IPv6Lab>show interface tunnel 0
Tunnel0 is up, line protocol is up
 Hardware is Tunnel
 Description: Becknet NOC IPv6 tunnel
 MTU 17920 bytes...
```

The display of "line protocol is up" indicates that the tunnel has been successfully connected. Next, try pinging your tunnel broker's server at the other end of the tunnel using the Server IPv6 address from the router command line.

```
IPv6Lab>ping 2001:470:1f0e:2e6::1
Sending 5, 100-byte ICMP Echos to 2001:470:1f0e:2e6::1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/4 ms
```

Now you're ready to plug a computer into your IPv6 lab network LAN. With Windows 7, which is fully IPv6 enabled, you should automatically get both an IPv6 and IPv4 address, as shown here.

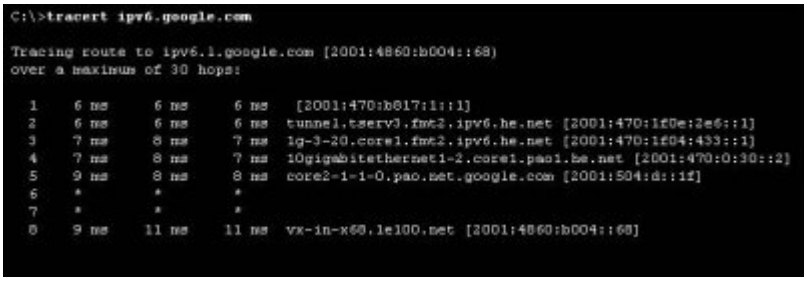
```
C:\>ipconfig /all

Windows IP Configuration
   Host Name . . . . . . . . . . . . : lenovo1
   Primary Dns Suffix  . . . . . . . :
   Node Type . . . . . . . . . . . . : Broadcast
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No

Ethernet adapter Local Area Connection:
   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Realtek RTL8139/810x Family Fast Ethernet NIC
   Physical Address. . . . . . . . . : 00-0F-B0-D5-93-AA
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   IPv6 Address. . . . . . . . . . . : 2001:470:b817:1:2194:5207:62db:b17e(Preferred)
   Temporary IPv6 Address. . . . . . : 2001:470:b817:1:51b7:3c72:886f:1b31(Preferred)
   Link-local IPv6 Address . . . . . : fe80::2194:5207:62db:b17e%12(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.6.11(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : Monday, November 16, 2009 1:25:36 PM
   Lease Expires . . . . . . . . . . : Tuesday, November 17, 2009 2:11:12 PM
   Default Gateway . . . . . . . . . : fe80::200:ff:fe01:0%12
                                       192.168.6.1
   DHCP Server . . . . . . . . . . . : 192.168.6.1
   DNS Servers . . . . . . . . . . . : 2001:570:20::2
                                       10.10.10.53
```

Click to expand.

If you successfully obtained an IPv6 address, try a tracert command from within Windows to verify that IPv6 networks are reachable, as shown in the first image below. If that works, then go ahead and surf the IPv6 Internet! Try my site at www.whatismyipv6.com first; you should get a response showing the IPv6 address you're coming from, as shown in the second image below. Next, try visiting various IPv6 destinations, such as ipv6.google.com. You're live on IPv6!



Click to expand.

Click to expand.

Down the Road

Getting your IPv6 lab functional is just the beginning. You'll want to start exploring various network tools, such as the DNS lookup utility nslookup, the IP path tracing tool traceroute, and the venerable ping utility, to see how they function with IPv6. We'll cover those tools, and more, in a future article on living with IPv6. For now, the more you explore, the more you'll learn!

Sidebar: Windows 7 and IPv6 Privacy

One of the unusual features of IPv6 in later Windows versions—Windows Vista, Windows Server 2008, and Windows 7—is the default use of random interface identifiers when creating an interface's IPv6 address. Under IPv6's standard Neighbor Discovery Protocol, an unconfigured device uses IPv6 autoconfiguration with the MAC address to form a 128-bit host address. The IETF's RFC 2373, "IP Version 6 Addressing Architecture," Appendix A, discusses the algorithm for deriving an EUI-64 based interface identifier from a MAC address. RFC 2464, "Transmission of IPv6 Packets over Ethernet Networks," Section 4, explains how stateless address autoconfiguration employs a device's MAC address. After these standards appeared, security experts expressed privacy concerns about using hardware MAC addresses as interface identifiers. Unlike NAT, which hides private addresses from Internet-based serves, a stock IPv6 address could be used to track individual users' activities over time. So the IETF issued RFC 4941 "Privacy Extensions for Stateless Address Autoconfiguration in IPv6," which defines a random interface address that changes over time, preserving the anonymity of individual devices on the Internet.

Windows 7 uses this randomizing technique by default, rather than the EUI-64 technique. This is a good thing for user privacy, but can complicate network troubleshooting and internal user activity tracking, because a device's IPv6 address can change with each new Internet connection. Fortunately, you can disable this behavior when necessary—at the cost of device anonymity, of course—using the command

```
netsh interface ipv6 set global randomizeidentifiers=disabled
```

Related Reading:

- Supporting IPv6 in Your Windows Server 2008 Environment
- IPv6: No Sticks, Just Carrots
- 2009: The Year of IPv6?
- Managing Your Migration and Transition from IPv4 to IPv6

Related Content:

- Q: How can I quickly find out my machine's IP address?
- The Evolution of Networking
- Managing Your Migration and Transition from IPv4 to IPv6
- Q. How do I use the command line to configure Windows Server 2008 and Windows Vista IPv4 static IP information?
- Calculated fields