



Step-by-Step Guide for Setting Up IPv6 in a Test Lab

Microsoft Corporation

Published: July, 2006

Author: Microsoft Corporation

Abstract

This guide describes how to configure Internet Protocol version 6 (IPv6) in a test lab using five computers. Of the five computers, one is a Domain Name System (DNS) server, two are clients, and two are routers. This guide also includes an exercise that disables IPv6 connectivity and then uses Intra-Site Automatic Tunnel Addressing Protocol (ISATAP) to restore it.

Microsoft

Information in this document, including URL and other Internet Web site references, is subject to change without notice. Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, e-mail address, logo, person, place, or event is intended or should be inferred. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 2006 Microsoft Corporation. All rights reserved.

Microsoft, MS-DOS, Windows, Windows NT, and Windows Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

All other trademarks are property of their respective owners.

Contents

Step-by-Step Guide for Setting Up IPv6 in a Test Lab	1
Abstract	1
Contents	3
Step-by-Step Guide for Setting Up IPv6 in a Test Lab	4
Setting up the IPv6 infrastructure	4
Consider using Virtual PC or Virtual Server	6
DNS1	6
CLIENT1	8
ROUTER1	11
ROUTER2	11
CLIENT2	12
Link-local addresses	13
Link-local ping	13
Creating a static IPv6 routing infrastructure	14
Global addresses	17
Global ping	17
Using name resolution	18
Configuring the test lab to use ISATAP	19
Conclusion	23
Additional references	23

Step-by-Step Guide for Setting Up IPv6 in a Test Lab

This guide provides detailed information about how you can use five computers to create a test lab with which to configure and test IPv6 with the Microsoft® Windows® XP Professional with Service Pack 2 (SP2) operating system and the 32-bit versions of the Windows Server™ 2003 with Service Pack 1 (SP1) operating system. These step-by-step instructions take you through the configuration required for setting up IPv6 in the test lab.

Note

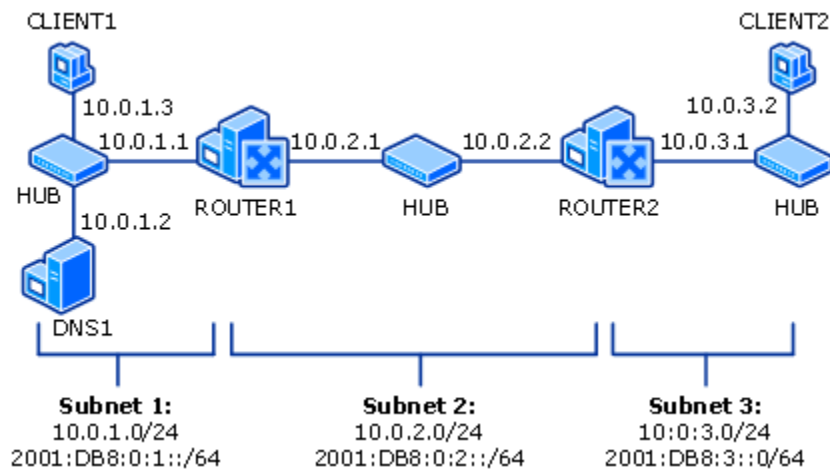
The following instructions are for configuring a test lab using a minimum number of computers. Individual computers are needed to separate the services provided on the network and to clearly show the desired functionality. This configuration is neither designed to reflect best practices nor is it designed to reflect a desired or recommended configuration for a production network. The configuration, including IP addresses and all other configuration parameters, is designed only to work on a separate test lab network.

Setting up the IPv6 infrastructure

The infrastructure for the IPv6 test lab network consists of five computers performing the following services:

- A computer running Windows Server 2003 with SP1, Standard Edition, that is used as a Domain Name System (DNS) server. This computer is named DNS1.
- A computer running Windows XP Professional with SP2 that is used as a client. This computer is named CLIENT1.
- A computer running Windows Server 2003 with SP1, Standard Edition, that is used as a router. This computer is named ROUTER1.
- A computer running Windows Server 2003 with SP1, Standard Edition, that is used as a router. This computer is named ROUTER2.
- A computer running Windows XP Professional with SP2 that is used as a client. This computer is named CLIENT2.

The following figure shows the configuration of the IPv6 test lab.



In the preceding figure, there are three network segments:

- Subnet 1 uses the private IP subnet prefix of 10.0.1.0/24 and global subnet prefix of 2001:DB8:0:1::/64.
- Subnet 2 uses the private IP subnet prefix of 10.0.2.0/24 and global subnet prefix of 2001:DB8:0:2::/64.
- Subnet 3 uses the private IP subnet prefix of 10.0.3.0/24 and global subnet prefix of 2001:DB8:3::/64.

All computers on each subnet are connected to a separate common hub or Layer 2 switch. The two router computers, ROUTER1 and ROUTER2, have two network adapters installed.

For the IPv4 configuration, each computer is manually configured with the appropriate IP address, subnet mask, default gateway, and DNS server IP address. For the IPv6 configuration, link-local addresses are used initially. Dynamic Host Configuration Protocol (DHCP) and Windows Internet Name Service (WINS) servers are not used.

The configuration of the lab routing infrastructure goes through three phases.

1. IPv4 routing. This phase demonstrates IPv4 connectivity and the automatic configuration of IPv6 local-link addresses - similar to Automatic Private IP Addressing (APIPA).
2. IPv6 routing. This phase demonstrates all test lab nodes can be reached by using IPv6 traffic.
3. ISATAP routing. ISATAP is an address assignment and automatic tunneling technology that is used to provide unicast IPv6 connectivity between IPv6/IPv4

hosts over an IPv4 intranet. This phase removes IPv6 connectivity for Subnet 2 and Subnet 3 and restores it using ISATAP.

The following sections describe how to configure each of the computers in the test lab with the basic IPv6 infrastructure. To create this test lab, configure the computers in the order presented.

Consider using Virtual PC or Virtual Server

Microsoft Virtual PC or Virtual Server allow you to create the computer lab used in this document using only one physical computer. After the virtual lab is configured, you can switch between the five virtual computers needed for this lab with the click of a button. For more information, see the following resources:

- Virtual PC 2004 Product Information
(<http://go.microsoft.com/fwlink/?LinkId=69217>)
- Virtual Server Product Information
(<http://go.microsoft.com/fwlink/?LinkId=69220>)
- Do More With Less: Exploring Virtual Server 2005
(<http://go.microsoft.com/fwlink/?LinkId=69221>)
- TechNet Webcast: Virtual Server 2005 - Setting Up a Virtual Test and Development Environment—Level 200
(<http://go.microsoft.com/fwlink/?LinkId=69222>)

DNS1

DNS1 is a computer running Windows Server 2003 with SP1, Standard Edition. It is providing DNS Server services for the testlab.microsoft.com DNS domain. To configure DNS1 for this service, perform the following steps.

▶ Configure DNS1 to provide DNS Services

1. Install Windows Server 2003 with SP1, Standard Edition, as a standalone server. Set the Administrator password.
2. After restarting, log on as Administrator.
3. Configure the TCP/IP protocol with the IP address of 10.0.1.2, the subnet mask of 255.255.255.0, and the default gateway of 10.0.1.1.

▶ **Install the DNS Server service**

1. Open Windows Components Wizard. To open the Windows Components Wizard, click **Start**, click Control Panel, double-click **Add or Remove Programs**, and then click **Add/Remove Windows Components**.
2. In **Components**, select the **Networking Services** check box, and then click **Details**.
3. In **Subcomponents of Networking Services**, select the **Domain Name System (DNS)** check box, click **OK**, and then click **Next**.
4. If prompted, in **Copy files from**, type the full path to the distribution files, and then click **OK**.

Define a forward lookup zone named testlab.microsoft.com that allows dynamic updates.

▶ **Define a forward lookup zone**

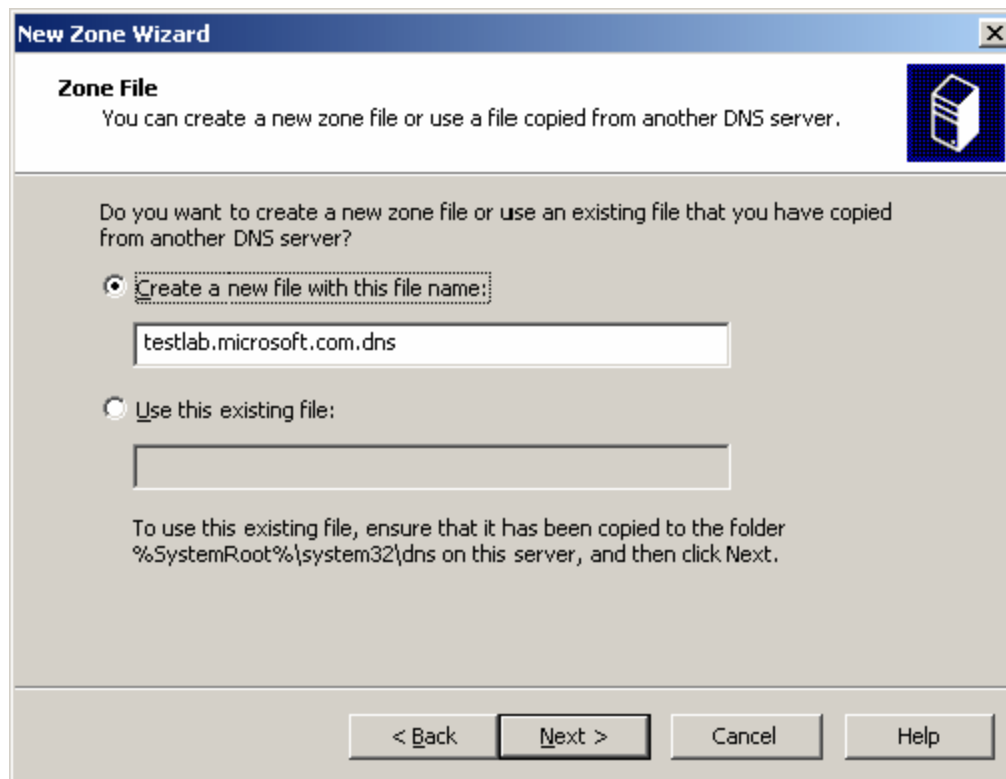
1. Open DNS. To open DNS, click **Start**, select **Administrative Tools**, and then click **DNS**.
2. In the console tree, click and then right-click the DNS server **DNS1**, and then click **New Zone** to run the New Zone Wizard.
3. On the **Welcome to the New Zone Wizard** page, click **Next**.
4. On the **Zone Type** page, the option to create a primary zone is selected by default. Click **Next**.
5. On the **Forward or Reverse Lookup Zone** page, the option to create a forward lookup zone is selected by default. Click **Next**.
6. On the **Zone Name** page, type **testlab.microsoft.com** in **Zone name**. Click **Next**.



Note

The domain name testlab.microsoft.com is used here for example purposes only. You can use any domain name in your test lab configuration.

7. On the **Zone File** page, the new zone file name, **testlab.microsoft.com.dns**, is automatically derived from the zone name entered on the previous page, as shown in the following figure. Click **Next**.



8. On the **Dynamic Update** page, select **Allow both nonsecure and secure dynamic updates**. Click **Next**.

9. Click **Next**. On the **Completing the New Zone Wizard** page, click **Finish**.

▶ Install IPv6

- On DNS1, at the command prompt, type:

```
netsh interface ipv6 install
```

CLIENT1

CLIENT1 is a computer that is being used as a client.

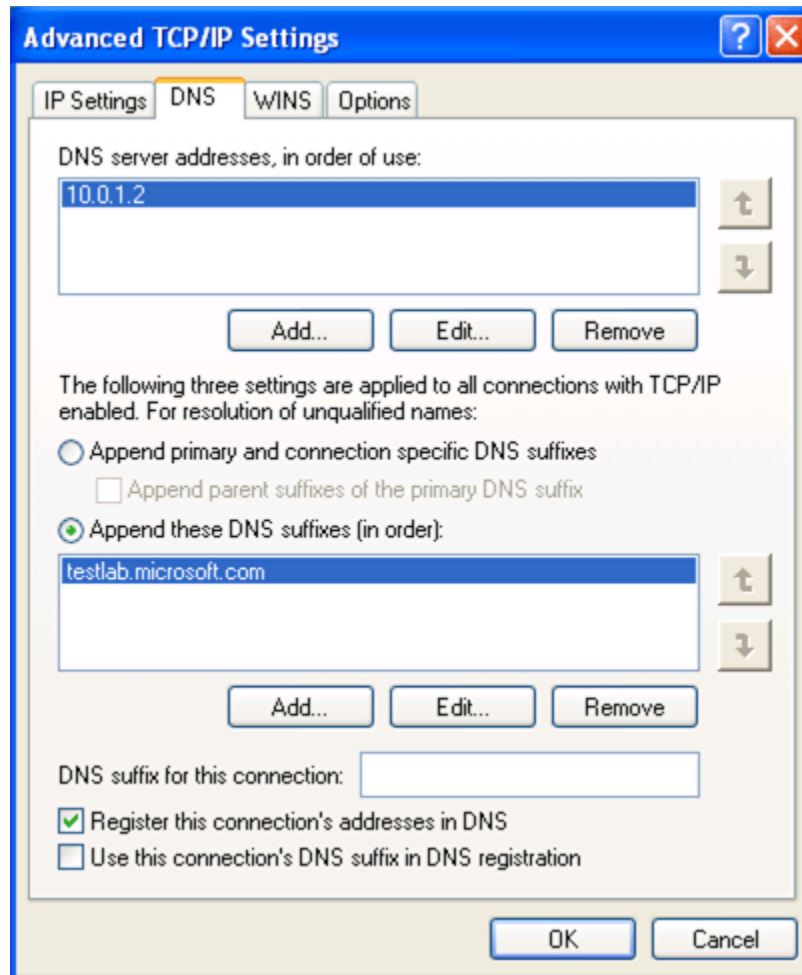
▶ Configure CLIENT1 as a client computer

1. Install Windows XP Professional with SP2 as a workgroup computer. Set the Administrator password.

2. After restarting, log on as Administrator.
3. At the command prompt, install the IPv6 protocol by typing:

netsh interface ipv6 install

4. Configure the TCP/IP protocol with the IP address of 10.0.1.3, the subnet mask of 255.255.255.0, a default gateway of 10.0.1.1, and the DNS server IP address of 10.0.1.2.
5. Append the DNS suffix **testlab.microsoft.com**. Click **Start**, click **Control Panel**, click **Network and Internet Connections**, and then click **Network Connections**. Right-click the network connection that you want to configure, and then click **Properties**. On the **General** tab, click **Internet Protocol (TCP/IP)**, and then click **Properties**. Click **Advanced**, and then click the **DNS** tab. Add **testlab.microsoft.com** to the list of DNS suffixes, as show in the following dialog box:



Windows Firewall is automatically turned on in Windows XP Professional with SP2. You need to configure an exception to allow CLIENT2 to ping CLIENT1.

▶ **Configure Windows to allow communication between client computers**

1. Click **Start**, point to **Control Panel**, and then click **Security Center**.
2. Click **Windows Firewall**, and then in the **Windows Firewall** dialog box, click the **Advanced** tab.
3. Click **Settings** for **ICMP**, and then click **Allow incoming echo request**.
4. Click **OK** twice to close **Windows Firewall**.

ROUTER1

ROUTER1 is a computer that is being used as a router between Subnet 1 and Subnet 2.

▶ Configure ROUTER1 as a router

1. Install Windows Server 2003 with SP1, Standard Edition, as a workgroup computer. Set the Administrator password.
2. After restarting, log on as Administrator.
3. At the command prompt, install the IPv6 protocol by typing:

netsh interface ipv6 install

4. In **Control Panel-Network Connections**, rename the LAN connection connected to Subnet 1 to **Subnet 1 Connection** and rename the LAN connection connected to Subnet 2 to **Subnet 2 Connection**.
5. For **Subnet 1 Connection**, configure the TCP/IP protocol with the IP address of 10.0.1.1, the subnet mask of 255.255.255.0, and the DNS server IP address of 10.0.1.2.
6. For **Subnet 2 Connection**, configure the TCP/IP protocol with the IP address of 10.0.2.1, the subnet mask of 255.255.255.0, and a default gateway of 10.0.2.2.
7. Start the registry editor (Regedit.exe) and set HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\IPEnableRouter to 1.

This step enables IPv4 routing between Subnet 1 and Subnet 2.

8. Restart the computer.

ROUTER2

ROUTER2 is a computer that is being used as a router between Subnet 2 and Subnet 3.

▶ Configure ROUTER2 as a router

1. Install Windows Server 2003 with SP1, Standard Edition, as a workgroup computer. Set the Administrator password.
2. After restarting, log on as Administrator.
3. At the command prompt, install the IPv6 protocol by typing:

netsh interface ipv6 install

4. Open Network Connections, and rename the LAN connection connected to Subnet 2 to **Subnet 2 Connection**, and rename the LAN connection connected to Subnet 3 to **Subnet 3 Connection**.
5. For **Subnet 2 Connection**, configure the TCP/IP protocol with the IP address of 10.0.2.2, the subnet mask of 255.255.255.0, and a default gateway of 10.0.2.1.
6. For **Subnet 3 Connection**, configure the TCP/IP protocol with the IP address of 10.0.3.1 and the subnet mask of 255.255.255.0.
7. Start the registry editor (Regedit.exe) and set HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\IPEnableRouter to 1.

This step enables IPv4 routing between Subnet 2 and Subnet 3.

8. Restart the computer.

CLIENT2

CLIENT2 is a computer that is being used as a client.

Configure CLIENT2 as a client computer

1. Install Windows XP Professional with SP2 as a workgroup computer. Set the Administrator password.
2. After restarting, log on as Administrator.
3. At the command prompt, install the IPv6 protocol by typing:

netsh interface ipv6 install

4. Configure the TCP/IP protocol with the IP address of 10.0.3.2, the subnet mask of 255.255.255.0, a default gateway of 10.0.3.1, and the DNS server IP address of 10.0.1.2, and append the DNS suffix **testlab.microsoft.com**.
5. Verify the integrity of the IPv4 routing infrastructure by pinging 10.0.1.3 from the CLIENT2 computer. On CLIENT2, type the following command:

ping 10.0.1.3

You should be able to successfully ping CLIENT1.

Link-local addresses

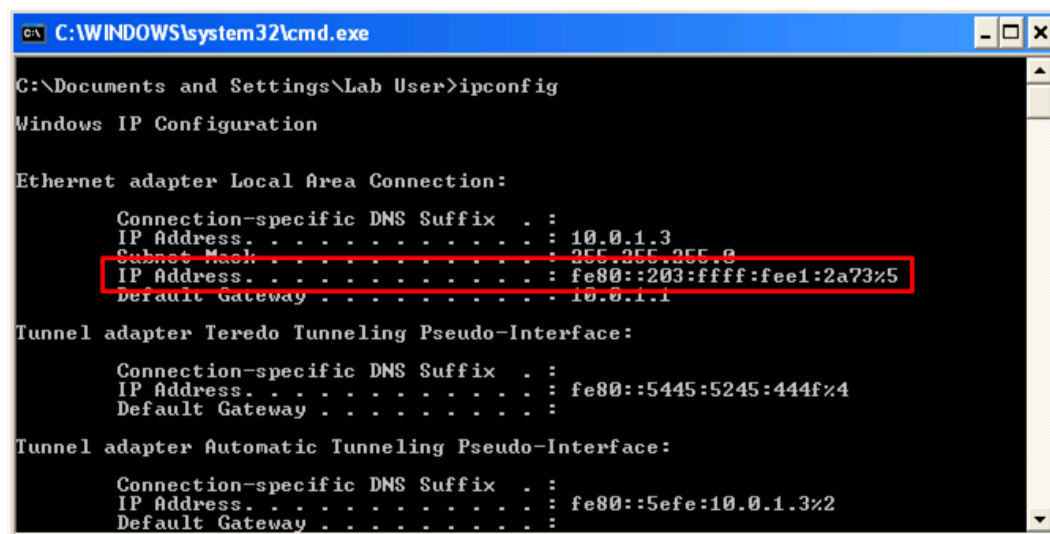
Link-local addresses, identified by the beginning address block FE80, are equivalent to APIPA IPv4 addresses autoconfigured on computers running current Microsoft Windows operating systems using the 169.254.0.0/16 prefix. One of the most useful aspects of IPv6 is its ability to automatically configure itself without the use of a stateful configuration protocol, such as Dynamic Host Configuration Protocol for IPv6 (DHCPv6). By default, an IPv6 node configures a link-local address for each interface. Link-local addresses are used by nodes when communicating with neighboring nodes on the same link (a network segment bounded by routers). The scope of a link-local address is the local link. An IPv6 router never forwards link-local traffic beyond the link.

Link-local ping

At this point in the lab configuration, only IPv4 traffic across the subnets is routed through an IPv4 routing infrastructure. However, you should be able to successfully ping neighboring link-local addresses. After the lab is configured to route IPv6 traffic, you will be able to successfully ping using global addresses.

► Determine and ping the link-local address of CLIENT1 from DNS1

1. On CLIENT1, type the **ipconfig** command to obtain the link-local address and interface index of the interface named Local Area Connection.



```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\Lab User>ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix  . : 
    IP Address. . . . .               : 10.0.1.3
    Subnet Mask . . . . .             : 255.255.255.0
    IP Address. . . . .               : fe80::203:ffff:fee1:2a73%5
    Default Gateway . . . . .         : 10.0.1.1

Tunnel adapter Teredo Tunneling Pseudo-Interface:

    Connection-specific DNS Suffix  . : 
    IP Address. . . . .               : fe80::5445:5245:444f%4
    Default Gateway . . . . .         : 

Tunnel adapter Automatic Tunneling Pseudo-Interface:

    Connection-specific DNS Suffix  . : 
    IP Address. . . . .               : fe80::5efe:10.0.1.3%2
    Default Gateway . . . . .         :
```

In this example, the link-local address is FE80::203:FFFF:FEE1:2A73 and the interface index is 5.

- On DNS1, type the **ipconfig** command to obtain the interface index of the interface named Local Area Connection.

In this example, the interface index of Local Area Connection is 4.

```

C:\Documents and Settings\Administrator>ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix  . : 
    IP Address. . . . .               : 10.0.1.2
    Subnet Mask . . . . .             : 255.255.255.0
    IP Address. . . . .               : fe80::203:ffff:fee4:fa73%4
    Default Gateway . . . . .         : 10.0.1.1

Tunnel adapter Teredo Tunneling Pseudo-Interface:

    Connection-specific DNS Suffix  . : 
    IP Address. . . . .               : fe80::5445:5245:444f%5
    Default Gateway . . . . .         : 

Tunnel adapter Automatic Tunneling Pseudo-Interface:

    Connection-specific DNS Suffix  . : 
    IP Address. . . . .               : fe80::5efe:10.0.1.2%2
    Default Gateway . . . . .         :
  
```

- On DNS1, ping the IPv6 link-local address of CLIENT1. You must use the interface index of the sending interface, the Local Area Connection interface of DNS1. For example, using the example configuration in step 1:

ping FE80::203:FFFF:FEE1:2A73%4

A successful ping demonstrates that CLIENT1 and DNS1 are neighbors on the same link.

Creating a static IPv6 routing infrastructure

Configure a static IPv6 routing infrastructure so that all test lab nodes can be reached by using IPv6 traffic.

► Create a static IPv6 routing infrastructure

- On ROUTER1, type the **ipconfig** command to obtain the link-local addresses of the interfaces connected to Subnet 1 Connection and Subnet 2 Connection.

```

C:\Documents and Settings\Administrator>ipconfig

Windows IP Configuration

Ethernet adapter Subnet 1 Connection:

    Connection-specific DNS Suffix  . : 
    IP Address . . . . . : 10.0.1.1
    Subnet Mask . . . . . : 255.255.255.0
    IP Address . . . . . : fe80::203:ffff:fee1:fa74%4
    Default Gateway . . . . . : 

Ethernet adapter Subnet 2 Connection:

    Connection-specific DNS Suffix  . : 
    IP Address . . . . . : 10.0.2.1
    Subnet Mask . . . . . : 255.255.255.0
    IP Address . . . . . : fe80::203:ffff:fefc:fa75%5
    Default Gateway . . . . . : 10.0.2.2

Tunnel adapter Teredo Tunneling Pseudo-Interface:

    Connection-specific DNS Suffix  . : 

```

For this example, the information is as follows:

Subnet 1 Connection - FE80::203:FFFF:FEE1:FA74

Subnet 2 Connection - FE80::203:FFFF:FEFC:FA75

- On ROUTER2, type the **ipconfig** command to obtain the link-local addresses of the interfaces connected to Subnet 2 Connection and Subnet 3 Connection.

```

C:\Documents and Settings\Administrator>ipconfig

Windows IP Configuration

Ethernet adapter Subnet 2 Connection:

    Connection-specific DNS Suffix  . : 
    IP Address . . . . . : 10.0.2.2
    Subnet Mask . . . . . : 255.255.255.0
    IP Address . . . . . : fe80::203:ffff:fee0:fa76%4
    Default Gateway . . . . . : 10.0.2.1

Ethernet adapter Subnet 3 Connection:

    Connection-specific DNS Suffix  . : 
    IP Address . . . . . : 10.0.3.1
    Subnet Mask . . . . . : 255.255.255.0
    IP Address . . . . . : fe80::203:ffff:fed:fa77%5
    Default Gateway . . . . . : 

Tunnel adapter Teredo Tunneling Pseudo-Interface:

    Connection-specific DNS Suffix  . : 

```

For this example, the information is as follows:

Subnet 2 Connection - FE80::203:FFFF:FEE0:FA76

Subnet 3 Connection - FE80::203:FFFF:FEFD:FA77

3. On ROUTER1, type the following commands:
 - a. **netsh interface ipv6 set interface "Subnet 1 Connection" forwarding=enabled advertise=enabled**
 - b. **netsh interface ipv6 set interface "Subnet 2 Connection" forwarding=enabled advertise=enabled**
 - c. **netsh interface ipv6 add route 2001:db8:0:1::/64 "Subnet 1 Connection" publish=yes**
 - d. **netsh interface ipv6 add route 2001:db8:0:2::/64 "Subnet 2 Connection" publish=yes**
 - e. **netsh interface ipv6 add route ::/0 "Subnet 2 Connection" nexthop=ROUTER2AddressOnSubnet2 publish=yes**

In the preceding command, *ROUTER2AddressOnSubnet2* represents the link-local address assigned to the Subnet 2 Connection interface on ROUTER2. For example, using the connection in this lab the command would be:

```
netsh interface ipv6 add route ::/0 "Subnet 2 Connection" nexthop=
FE80::203:FFFF:FEE0:FA76 publish=yes
```

4. On ROUTER2, type the following commands:
 - a. **netsh interface ipv6 set interface "Subnet 2 Connection" forwarding=enabled advertise=enabled**
 - b. **netsh interface ipv6 set interface "Subnet 3 Connection" forwarding=enabled advertise=enabled**
 - c. **netsh interface ipv6 add route 2001:db8:0:2::/64 "Subnet 2 Connection" publish=yes**
 - d. **netsh interface ipv6 add route 2001:db8:0:3::/64 "Subnet 3 Connection" publish=yes**
 - e. **netsh interface ipv6 add route ::/0 "Subnet 2 Connection" nexthop=ROUTER1AddressOnSubnet2 publish=yes**

In the preceding command, *ROUTER1AddressOnSubnet2* represents the link-local address assigned to the Subnet 2 Connection interface on ROUTER1. For example, using the connection in this lab the command would be:

```
netsh interface ipv6 add route ::/0 "Subnet 2 Connection" nexthop=
FE80::203:FFFF:FEFC:FA75 publish=yes
```

Global addresses

At this point, you have now created static IPv6 routes with global subnet prefixes. Global addresses are equivalent to the IPv4 public address space. Unlike link-local addresses, global addresses are not automatically configured and must be assigned either through stateless or stateful address configuration processes. You can determine which address is the global address by the "ff:fe" portion of the public address in the sixth and seventh blocks of the address, indicating an EUI-64-based interface ID. The Extended Unique Identifier (EUI)-64 address is a newer 64-bit MAC address.

Global ping

► Verify the IPv6 routing structure

1. On CLIENT1, type the **ipconfig** command to check for a new global IPv6 address.

```

C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\Lab User>ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix  . : 
    IP Address. . . . .               : 10.0.1.3
    Subnet Mask . . . . .             : 255.255.255.0
    IP Address. . . . .               : 2001:db8:0:1:347e:9198:e536:e75
    IP Address. . . . .               : 2001:db8:0:1:203:ffff:fee1:2a73
    IP Address. . . . .               : fe80::203:ffff:fee1:2a73%5
    Default Gateway . . . . .         : 10.0.1.1
                                         fe80::203:ffff:fee1:fa74%5

Tunnel adapter Teredo Tunneling Pseudo-Interface:

    Connection-specific DNS Suffix  . : 
    IP Address. . . . .               : fe80::5445:5245:444f%4
    Default Gateway . . . . .         : 

Tunnel adapter Automatic Tunneling Pseudo-Interface:
  
```

In this example, the CLIENT1 global address is 2001:DB8:0:1:203:FFFF:FEE1:2A73.

2. On CLIENT2, type the following commands:

ping CLIENT1GlobalAddress

tracert -d CLIENT1GlobalAddress

In this example, the CLIENT1 global address is 2001:DB8:0:1:203:FFFF:FEE1:2A73.

A successful ping and tracert demonstrates that IPv6 static routes have been created

and are functioning.

3. You can view the entries in the ROUTER1 neighbor cache for CLIENT1 and ROUTER2, by typing the following on ROUTER1:

```
netsh interface ipv6 show neighbors
```

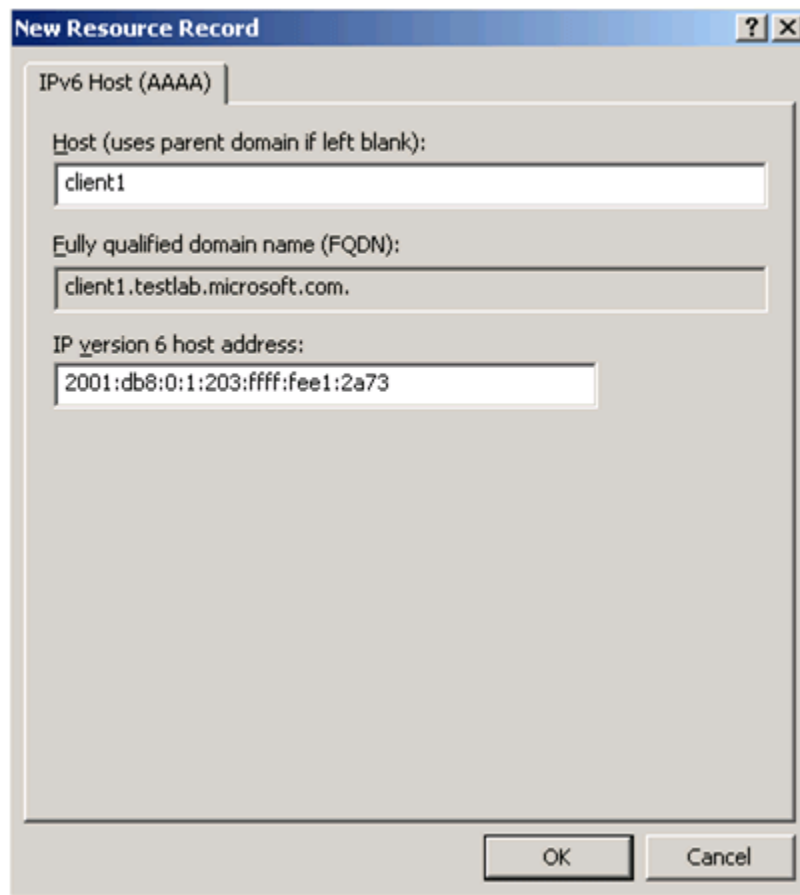
Using name resolution

To resolve host names to IPv6 addresses, you must first configure DNS. On DNS1, create an AAAA record for CLIENT2 with the DNS name `client2.testlab.microsoft.com` for its global IPv6 address using the IPv6 Host resource record type.

Configure DNS to resolve names to IPv6 addresses

1. On DNS1, click **Start**, select **Administrative Tools**, and then click **DNS**.
2. In the console tree, click and then right-click **testlab.microsoft.com** in the **Forward Lookup Zones** folder, and then click **Other New Records**.
3. Click **IPv6 Host (AAAA)**, and then click **Create Record**.
4. In the **Host** text box type **client1**, and then in the **IP version 6 host address** text box, enter the IPv6 global address for CLIENT1.

In this example, the CLIENT1 global address is
2001:DB8:0:1:203:FFFF:FEE1:2A73.



The screenshot shows a Windows-style dialog box titled "New Resource Record" with a tab labeled "IPv6 Host (AAAA)". It contains three text input fields. The first field is labeled "Host (uses parent domain if left blank):" and contains the text "client1". The second field is labeled "Fully qualified domain name (FQDN):" and contains "client1.testlab.microsoft.com.". The third field is labeled "IP version 6 host address:" and contains "2001:db8:0:1:203:fff:fee1:2a73". At the bottom right of the dialog are two buttons: "OK" and "Cancel".

5. Click **OK**, and then click **Done** to add the new IPv6 host record to the zone.
6. On CLIENT 2, type the following command:

ping client1

A successful ping demonstrates that host names can resolve to IPv6 addresses.

Configuring the test lab to use ISATAP

ISATAP is an address assignment and automatic tunneling technology that is used to provide unicast IPv6 connectivity between IPv6/IPv4 hosts on an IPv4 intranet, which are also known as ISATAP hosts. ISATAP is described in RFC 4214. More information about ISATAP can be found in the IPv6 Transition Technologies white paper (<http://go.microsoft.com/fwlink/?LinkId=67210>). By default, the IPv6 protocol for

Windows XP Professional with SP2 and Windows Server 2003 with SP1, Standard Edition, configures a link-local ISATAP address on the Automatic Tunneling Pseudo-Interface for each IPv4 address assigned to a computer.

To configure global ISATAP addresses, or to communicate beyond the logical subnet defined by the IPv4 intranet, you need an ISATAP router. An ISATAP router performs the following functions:

- Advertises its presence and address prefixes, enabling global ISATAP addresses to be configured.
- Optionally forwards IPv6 packets between ISATAP hosts on the IPv4 intranet and IPv6 hosts beyond it.

An ISATAP router is typically configured to perform both functions, but can perform either individually. Most often, an ISATAP router acts as the forwarder between ISATAP hosts on an IPv4 intranet and IPv6 hosts on an IPv6-enabled portion of an intranet.

To demonstrate the use of an ISATAP router between IPv6 and IPv4 intranets, the following steps first separate the lab into a portion that has IPv4 and IPv6 connectivity and another that has IPv4 connectivity only. Then, ROUTER1 is configured as an ISATAP router so that hosts on the IPv4-only portion of the intranet can communicate with hosts on the IPv6-enabled portion of the intranet.

To do this, we will disable IPv6 forwarding and advertising on the Subnet 2 Connection interface of ROUTER1 and both interfaces of ROUTER2. This emulates an intranet in which a portion is IPv6-enabled (Subnet 1) and a portion is not (Subnet 2 and Subnet 3).

► Configure the test lab to use ISATAP

1. Disable forwarding and advertising on the Subnet 2 Connection interface of ROUTER1 and ROUTER2.

a. On ROUTER1, at the command prompt, type:

```
netsh interface ipv6 set interface "Subnet 2 Connection"  
forwarding=disabled advertise=disabled
```

b. On ROUTER2, at the command prompt, type:

```
netsh interface ipv6 set interface "Subnet 2 Connection"  
forwarding=disabled advertise=disabled
```

2. Disable forwarding and advertising on the Subnet 3 Connection interface. On ROUTER2, at the command prompt, type:

```
netsh interface ipv6 set interface "Subnet 3 Connection"  
forwarding=disabled advertise=disabled
```

3. Renew CLIENT2 configuration to remove the global address and route. On CLIENT2, type the following command:

```
netsh interface ipv6 renew
```

4. Ping CLIENT1 from CLIENT2. On CLIENT2, type the following command:

```
ping CLIENT1GlobalAddress
```

or

```
ping client1
```

In this example, the CLIENT1 global address is 2001:DB8:0:1:203:FFFF:FEE1:2A73.

This Ping command fails because there is no longer IPv6 connectivity between CLIENT2 and CLIENT1.

5. Configure ROUTER1 as an ISATAP router advertising the 2001:DB8:0:10::/64 route to the logical ISATAP subnet consisting of Subnet 2 and Subnet 3.

- a. On ROUTER1, enable forwarding and advertising on the Automatic Tunneling Pseudo-Interface, using the following commands:

```
netsh interface ipv6 isatap set router 10.0.2.1
```

```
netsh interface ipv6 set interface "Automatic Tunneling Pseudo-Interface" forwarding=enabled advertise=enabled
```

- b. On ROUTER1, add a route for the subnet prefix of the logical subnet of the IPv4 intranet to the Automatic Tunneling Pseudo-Interface and configure it to be published. Use the following command:

```
netsh interface ipv6 add route 2001:db8:0:10::/64 "Automatic Tunneling Pseudo-Interface" publish=yes
```

6. Add an ISATAP address (A) resource record to the DNS server for the IPv4 address of 10.0.2.1.
 - a. On DNS1, click **Start**, select **Administrative Tools**, and then click **DNS**.
 - b. In the console tree, click and then right-click **testlab.microsoft.com** in the **Forward Lookup Zones** folder, and

then click **New Host (A)**.

c. In the **Name** text box, type **ISATAP**, and in the **IP address** text box, type **10.0.2.1**, as shown in the following figure.

The screenshot shows a 'New Host' dialog box with the following fields and values:

- Name (uses parent domain name if blank):** ISATAP
- Fully qualified domain name (FQDN):** ISATAP.testlab.microsoft.com.
- IP address:** 10 .0 .2 .1
- Create associated pointer (PTR) record:**

Buttons at the bottom: Add Host, Cancel.

d. Click **Add Host**, click **OK**, and then click **Done** to add the new host record to the zone.

7. Renew the CLIENT2 configuration to recognize the ISATAP router IPv4 address of ROUTER1 and obtain an ISATAP address with the global prefix of 2001:DB8:0:10::/64. On CLIENT2, type the following command:

netsh interface ipv6 renew

8. Ping CLIENT1 from CLIENT2. On CLIENT2, type the following command:

ping CLIENT1GlobalAddress

or

ping client1

In this example, the CLIENT1 global address is 2001:DB8:0:1:203:FFFF:FEE1:2A73.

This ping command succeeds because IPv6 connectivity using ISATAP now

exists between CLIENT2 and CLIENT1. On the link between CLIENT1 and ROUTER1, the IPv6 traffic is encapsulated as IPv4 packets.

 **Note**

It may take a few minutes for IPv6 connectivity between CLIENT2 and CLIENT1 to be established. You may need to restart CLIENT2 and run **netsh interface ipv6 renew** on CLIENT1.

Conclusion

This guide described the steps required to configure IPv6 in a test lab with five computers. It also described how to use ISATAP to provide IPv6 connectivity between IPv6/IPv4 hosts on an IPv4-only portion of an intranet and IPv6 hosts on an IPv6-enabled portion of an intranet.

Additional references

For more information, see the following resources:

- Microsoft IPv6 Web site (<http://go.microsoft.com/fwlink/?LinkId=24350>)
- Introduction to IPv6 (<http://go.microsoft.com/fwlink/?LinkId=69223>)
- Support WebCast: Introduction to Internet Protocol Version 6 (<http://go.microsoft.com/fwlink/?LinkId=69224>)
- Using Windows Tools to Obtain IPv6 Configuration Information (<http://go.microsoft.com/fwlink/?LinkId=69225>)
- Windows Server 2003 Web site (<http://go.microsoft.com/fwlink/?LinkId=5221>)