Contents

Configuring DNS

About DNS

Domain Name System (DNS) is a distributed database used by TCP/IP applications to translate domain names into IP addresses. The domain name-to-IP address mapping is called a DNS entry.

Types of DNS services

DNS services can be static or dynamic. After a user specifies a name, the device checks the static name resolution table for an IP address. If no IP address is available, it contacts the DNS server for dynamic name resolution, which takes more time than static name resolution. To improve efficiency, you can put frequently queried name-to-IP address mappings in the local static name resolution table.

Static domain name resolution

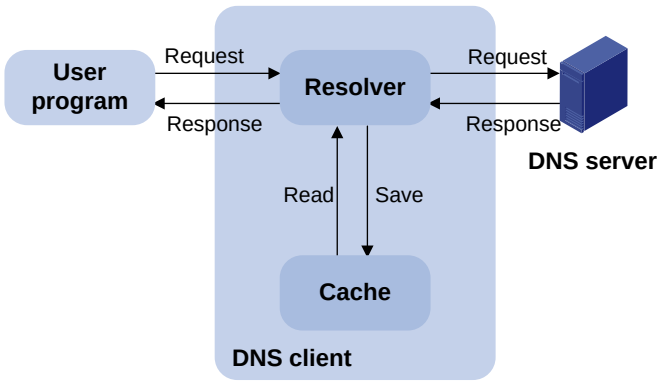
Static domain name resolution means manually creating mappings between domain names and IP addresses. For example, you can create a static DNS mapping for a device so that you can Telnet to the device by using the domain name.

Dynamic domain name resolution

Architecture

Figure 1 shows the relationship between the user program, DNS client, and DNS server. The DNS client includes the resolver and cache. The user program and DNS client can run on the same device or different devices. The DNS server and the DNS client usually run on different devices.

Figure 1 Dynamic domain name resolution



The device can function as a DNS client, but not a DNS server.

If an alias is configured for a domain name on the DNS server, the device can resolve the alias into the IP address of the host.

Resolution process

The dynamic domain name resolution process is as follows:

1. A user program sends a name query to the resolver of the DNS client.

2. The DNS resolver looks up the local domain name cache for a match. If the resolver finds a match, it sends the corresponding IP address back. If not, it sends a query to the DNS server.
3. The DNS server looks up the corresponding IP address of the domain name in its DNS database. If no match is found, the server sends a query to other DNS servers. This process continues until a result, whether successful or not, is returned.
4. After receiving a response from the DNS server, the DNS client returns the resolution result to the user program.

Caching

Dynamic domain name resolution allows the DNS client to store latest DNS entries in the DNS cache. The DNS client does not need to send a request to the DNS server for a repeated query within the aging time. To make sure the entries from the DNS server are up to date, a DNS entry is removed when its aging timer expires. The DNS server determines how long a mapping is valid, and the DNS client obtains the aging information from DNS responses.

DNS suffixes

You can configure a domain name suffix list so that the resolver can use the list to supply the missing part of an incomplete name.

For example, you can configure **com** as the suffix for aabbcc.com. The user only needs to enter **aabbcc** to obtain the IP address of aabbcc.com. The resolver adds the suffix and delimiter before passing the name to the DNS server.

The name resolver handles the queries based on the domain names that the user enters:

- If the user enters a domain name without a dot (.) (for example, aabbcc), the resolver considers the domain name to be a host name. It adds a DNS suffix to the host name before performing the query operation. If no match is found for any host name and suffix combination, the resolver uses the user-entered domain name (for example, aabbcc) for the IP address query.
- If the user enters a domain name with a dot (.) among the letters (for example, www.aabbcc), the resolver directly uses this domain name for the query operation. If the query fails, the resolver adds a DNS suffix for another query operation.
- If the user enters a domain name with a dot (.) at the end (for example, aabbcc.com.), the resolver considers the domain name an FQDN and returns the successful or failed query result. The dot at the end of the domain name is considered a terminating symbol.

DNS proxy

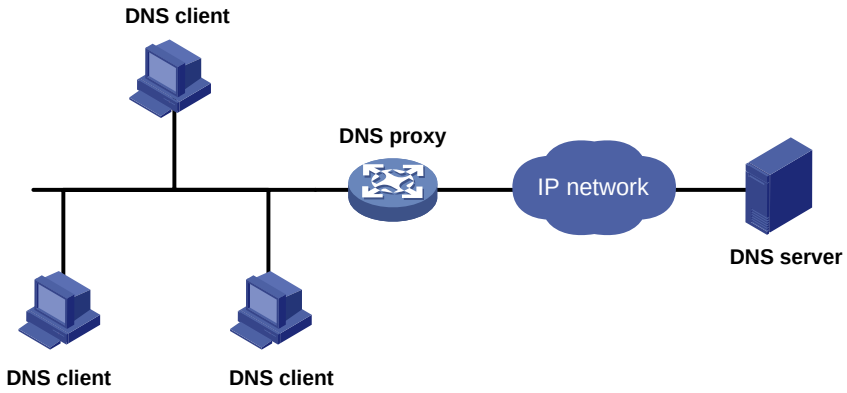
The DNS proxy performs the following functions:

- Forwards the request from the DNS client to the designated DNS server.
- Conveys the reply from the DNS server to the client.

The DNS proxy simplifies network management. When the DNS server address is changed, you can change the configuration only on the DNS proxy instead of on each DNS client.

Figure 2 shows the typical DNS proxy application.

Figure 2 DNS proxy application



A DNS proxy operates as follows:

1. A DNS client considers the DNS proxy as the DNS server, and sends a DNS request to the DNS proxy. The destination address of the request is the IP address of the DNS proxy.
2. The DNS proxy searches the local static domain name resolution table and dynamic domain name resolution cache after receiving the request. If the requested information is found, the DNS proxy returns a DNS reply to the client.
3. If the requested information is not found, the DNS proxy sends the request to the designated DNS server for domain name resolution.
4. After receiving a reply from the DNS server, the DNS proxy records the IP address-to-domain name mapping and forwards the reply to the DNS client.

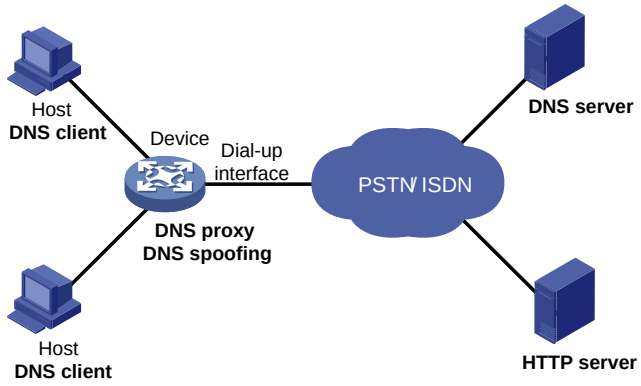
If no DNS server is designated or no route is available to the designated DNS server, the DNS proxy does not forward DNS requests.

DNS spoofing

As shown in Figure 3, DNS spoofing is applied to the dial-up network.

- The device connects to a PSTN/ISDN network through a dial-up interface. The device triggers the establishment of a dial-up connection only when packets are to be forwarded through the dial-up interface.
- The device acts as a DNS proxy and is specified as a DNS server on the hosts. After the dial-up connection is established, the device dynamically obtains the DNS server address through DHCP or another autoconfiguration mechanism.

Figure 3 DNS spoofing application



The DNS proxy does not have the DNS server address or cannot reach the DNS server after startup. A host accesses the HTTP server in the following steps:

1. The host sends a DNS request to the device to resolve the domain name of the HTTP server into an IP address.
2. Upon receiving the request, the device searches the local static and dynamic DNS entries for a match. Because no match is found, the device spoofs the host by replying a configured IP address. The device must have a route to the IP address with the dial-up interface as the output interface.

   The IP address configured for DNS spoofing is not the actual IP address of the requested domain name. Therefore, the TTL field is set to 0 in the DNS reply. When the DNS client receives the reply, it creates a DNS entry and ages it out immediately.
3. Upon receiving the reply, the host sends an HTTP request to the replied IP address.
4. When forwarding the HTTP request through the dial-up interface, the device performs the following operations:
   - Establishes a dial-up connection with the network.
   - Dynamically obtains the DNS server address through DHCP or another autoconfiguration mechanism.
5. Because the DNS entry ages out immediately upon creation, the host sends another DNS request to the device to resolve the HTTP server domain name.
6. The device operates the same as a DNS proxy. For more information, see "DNS proxy."
7. After obtaining the IP address of the HTTP server, the host can access the HTTP server.

Without DNS spoofing, the device forwards the DNS requests from the host to the DNS server if it cannot find a matching local DNS entry. However, the device cannot obtain the DNS server address, because no dial-up connection is established. Therefore, the device cannot forward or answer the requests from the client. DNS resolution fails, and the client cannot access the HTTP server.

DNS tasks at a glance

To configure DNS, perform the following tasks:

1. Configuring the DNS client

   Choose the following tasks as needed:
   - Configuring static domain name resolution
   - Configuring dynamic domain name resolution
2. (Optional.) Configuring the DNS proxy
3. (Optional.) Configuring DNS spoofing

   This feature is applied to the dial-up network.
4. (Optional.) Specifying the source interface for DNS packets
5. (Optional.) Configuring the DNS trusted interface
6. (Optional.) Setting the DSCP value for outgoing DNS packets

Configuring the DNS client

Configuring static domain name resolution

Restrictions and guidelines

For the public network or a VPN instance, each host name maps to only one IPv4 address and one IPv6 address.

A maximum of 2048 DNS entries can be configured for the public network or each VPN instance. You can configure DNS entries for both public network and VPN instances.

Procedure

1. Enter system view.

   system-view

2. Configure a host name-to-address mapping. Choose the options to configure as needed:

   IPv4:

   ip host *host-name ip-address* [**vpn-instance** *vpn-instance-name*]

   IPv6:

   ipv6 host *host-name ipv6-address* [**vpn-instance** *vpn-instance-name*]

Configuring dynamic domain name resolution

Restrictions and guidelines

- The limit on the number of DNS servers on the device is as follows:
  - In system view, you can specify a maximum of six DNS server IPv4 addresses for the public network or each VPN instance. You can specify DNS server IPv4 addresses for both public network and VPN instances.
  - In system view, you can specify a maximum of six DNS server IPv6 addresses for the public network or each VPN instance. You can specify DNS server IPv6 addresses for both public network and VPN instances.
  - In interface view, you can specify a maximum of six DNS server IPv4 addresses for the public network or each VPN instance. You can specify DNS server IPv4 addresses for both public network and VPN instances.
- A DNS server address is required so that DNS queries can be sent to a correct server for resolution. If you specify both an IPv4 address and an IPv6 address, the device performs the following operations:
  - Sends an IPv4 DNS query first to the DNS server IPv4 addresses. If the query fails, the device turns to the DNS server IPv6 addresses.
  - Sends an IPv6 DNS query first to the DNS server IPv6 addresses. If the query fails, the devices turns to the DNS server IPv4 addresses.
- A DNS server address specified in system view takes priority over a DNS server address specified in interface view. A DNS server address specified earlier has a higher priority. A DNS server address manually specified takes priority over a DNS server address dynamically obtained, for example, through DHCP. The device first sends a DNS query to the DNS server address of the highest priority. If the first query fails, it sends the DNS query to the DNS server address of the second highest priority, and so on.
- You can configure a DNS suffix that the system automatically adds to the incomplete domain name that a user enters.
  - You can configure a maximum of 16 DNS suffixes for the public network or each VPN instance. You can configure DNS suffixes for both public network and VPN instances.
  - A DNS suffix manually configured takes priority over a DNS suffix dynamically obtained, for example, through DHCP. A DNS suffix configured earlier has a higher priority. The device first uses the suffix that has the highest priority. If the query fails, the device uses the suffix that has the second highest priority, and so on.

Procedure

1. Enter system view.

   system-view

2. (Optional.) Configure a DNS suffix.

   dns domain *domain-name* [**vpn-instance** *vpn-instance-name*]

   By default, no DNS suffix is configured and only the domain name that a user enters is resolved.

3. Specify a DNS server address.

   IPv4:

   dns server *ip-address* [**vpn-instance** *vpn-instance-name*]

   IPv6:

   ipv6 dns server *ipv6-address* [*interface-type interface-number*] [**vpn-instance** *vpn-instance-name*]

   By default, no DNS server address is specified.

Configuring the DNS proxy

Restrictions and guidelines

You can specify multiple DNS servers. The DNS proxy forwards a request to the DNS server that has the highest priority. If having not received a reply, it forwards the request to a DNS server that has the second highest priority, and so on.

You can specify both an IPv4 address and an IPv6 address.

- A DNS proxy forwards an IPv4 name query first to IPv4 DNS servers. If no reply is received, it forwards the request to IPv6 DNS servers.
- A DNS proxy forwards an IPv6 name query first to IPv6 DNS servers. If no reply is received, it forwards the request to IPv4 DNS servers.

Procedure

1. Enter system view.

   system-view

2. Enable DNS proxy.

   dns proxy enable

   By default, DNS proxy is disabled.

3. Specify a DNS server address.

   IPv4:

   dns server *ip-address* [**vpn-instance** *vpn-instance-name*]

   IPv6:

   ipv6 dns server *ipv6-address* [*interface-type interface-number*] [**vpn-instance** *vpn-instance-name*]

   By default, no DNS server address is specified.

Configuring DNS spoofing

Restrictions and guidelines

- You can configure only one replied IPv4 address and one replied IPv6 address for the public network or a VPN instance. If you execute the command multiple times, the most recent configuration takes effect.
- You can configure DNS spoofing for both public network and VPN instances.
- After DNS spoofing takes effect, the device spoofs a DNS request even though a matching static DNS entry exists.

Prerequisites

The DNS proxy is enabled on the device.

No DNS server or route to any DNS server is specified on the device.

Procedure

1. Enter system view.

   system-view

2. Enable DNS proxy.

   dns proxy enable

   By default, DNS proxy is disabled.

3. Enable DNS spoofing and specify the IP address used to spoof DNS requests. Choose one option as needed:

   IPv4:

   dns spoofing *ip-address* [**vpn-instance** *vpn-instance-name*]

   IPv6:

   ipv6 dns spoofing *ipv6-address* [**vpn-instance** *vpn-instance-name*]

   By default, DNS spoofing is disabled.

Specifying the source interface for DNS packets

About the source interface for DNS packets

This task enables the device to always use the primary IP address of the specified source interface as the source IP address of outgoing DNS packets. This feature applies to scenarios in which the DNS server responds only to DNS requests sourced from a specific IP address. If no IP address is configured on the source interface, no DNS packets can be sent out.

Restrictions and guidelines

When sending an IPv6 DNS request, the device follows the method defined in RFC 3484 to select an IPv6 address of the source interface.

You can configure only one source interface on the public network or a VPN instance. You can configure source interfaces for both public network and VPN instances.

Make sure the source interface belongs to the specified VPN instance if you specify the **vpn-instance** *vpn-instance-name* option.

Procedure

1. Enter system view.

   system-view

2. Specify the source interface for DNS packets.

   dns source-interface *interface-type interface-number* [**vpn-instance** *vpn-instance-name*]

   By default, no source interface for DNS packets is specified.

Configuring the DNS trusted interface

About DNS trusted interface

This task enables the device to use only the DNS suffix and domain name server information obtained through the trusted interface. The device can then obtain the correct resolved IP address.

This feature protects the device against attackers that act as the DHCP server to assign incorrect DNS suffix and domain name server address.

Restrictions and guidelines

You can configure a maximum of 128 DNS trusted interfaces.

Procedure

1. Enter system view.
   system-view
2. Specify the DNS trusted interface.
   dns trust-interface *interface-type interface-number*
   By default, no DNS trusted interface is specified.

Setting the DSCP value for outgoing DNS packets

About the DSCP value for outgoing DNS packets

The DSCP value of a packet specifies the priority level of the packet and affects the transmission priority of the packet. A bigger DSCP value represents a higher priority.

Procedure

1. Enter system view.
   system-view
2. Set the DSCP value for DNS packets sent by a DNS client or a DNS proxy.
   IPv4:
   dns dscp *dscp-value*
   By default, the DSCP value is 0 in IPv4 DNS packets sent by a DNS client or a DNS proxy.
   IPv6:
   ipv6 dns dscp *dscp-value*
   By default, the DSCP value is 0 in IPv6 DNS packets sent by a DNS client or a DNS proxy.

Display and maintenance commands for DNS

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display DNS suffixes.	**display dns domain** [**dynamic**] [**vpn-instance** *vpn-instance-name*]
Display the domain name resolution table.	**display dns host** [**ip** | **ipv6**] [**vpn-instance** *vpn-instance-name*]
Display IPv4 DNS server information.	**display dns server** [**dynamic**] [**vpn-instance** *vpn-instance-name*]
Display IPv6 DNS server information.	**display ipv6 dns server** [**dynamic**] [**vpn-instance** *vpn-instance-name*]
Clear dynamic DNS entries.	**reset dns host** [**ip** | **ipv6**] [**vpn-instance** *vpn-instance-name*]

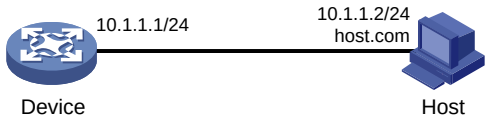
IPv4 DNS configuration examples

Example: Configuring static domain name resolution

Network configuration

As shown in Figure 4, the host at 10.1.1.2 is named **host.com**. Configure static IPv4 DNS on the device so that the device can use the easy-to-remember domain name rather than the IP address to access the host.

Figure 4 Network diagram



Procedure

Configure a mapping between host name **host.com** and IP address 10.1.1.2.

```
<Sysname> system-view
[Sysname] ip host host.com 10.1.1.2
```

Verify that the device can use static domain name resolution to resolve domain name **host.com** into IP address 10.1.1.2.

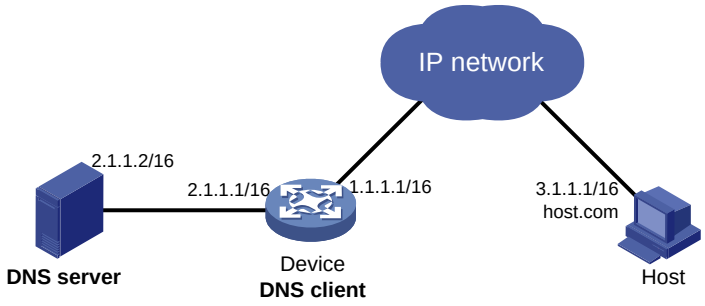
```
[Sysname] ping host.com
Ping host.com (10.1.1.2): 56 data bytes, press CTRL_C to break
56 bytes from 10.1.1.2: icmp_seq=0 ttl=255 time=1.000 ms
56 bytes from 10.1.1.2: icmp_seq=1 ttl=255 time=1.000 ms
56 bytes from 10.1.1.2: icmp_seq=2 ttl=255 time=1.000 ms
56 bytes from 10.1.1.2: icmp_seq=3 ttl=255 time=1.000 ms
56 bytes from 10.1.1.2: icmp_seq=4 ttl=255 time=2.000 ms

--- Ping statistics for host.com ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 1.000/1.200/2.000/0.400 ms
```

Example: Configuring dynamic domain name resolution

Network configuration

As shown in Figure 5, configure the DNS server to store the mapping between the host's domain name **host** and IPv4 address 3.1.1.1/16 in the **com** domain. Configure dynamic IPv4 DNS and DNS suffix **com** on the device so that the device can use domain name **host** to access the host.

Figure 5 Network diagram



Procedure

Before performing the following configuration, make sure that:

- The device and the host can reach each other.
- The IP addresses of the interfaces are configured as shown in Figure 5.

1. Configure the DNS server:

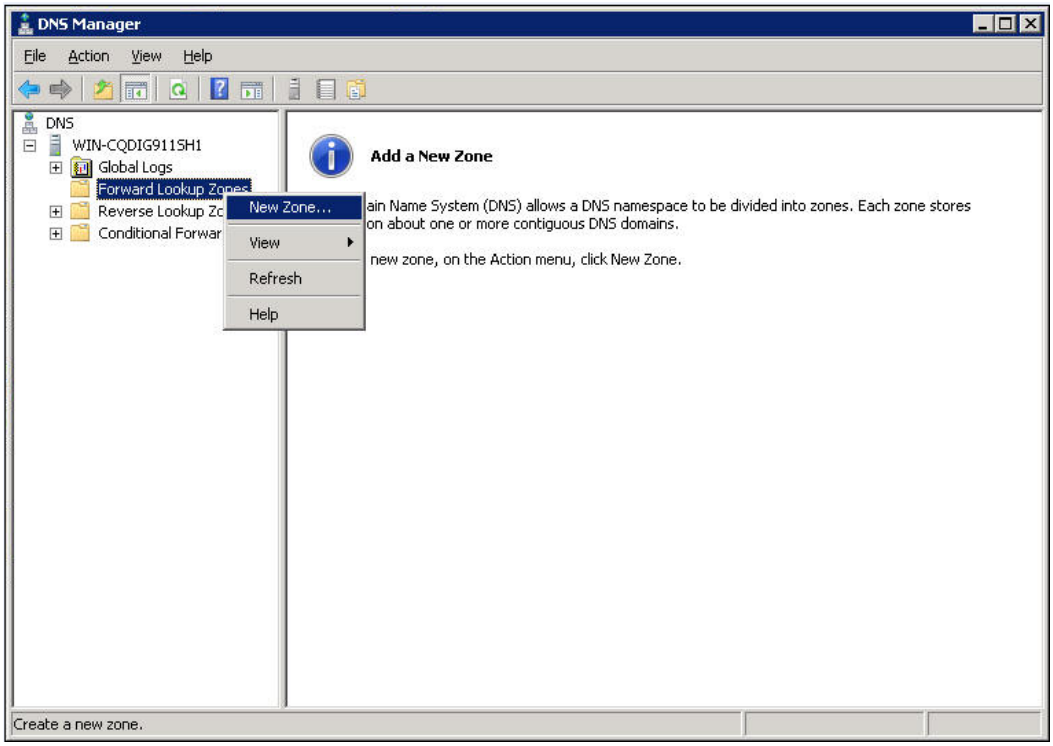
The configuration might vary by DNS server. The following configuration is performed on a PC running Windows Server 2008 R2.

a. Select **Start** > **Programs** > **Administrative Tools** > **DNS**.

The DNS server configuration page appears, as shown in Figure 6.

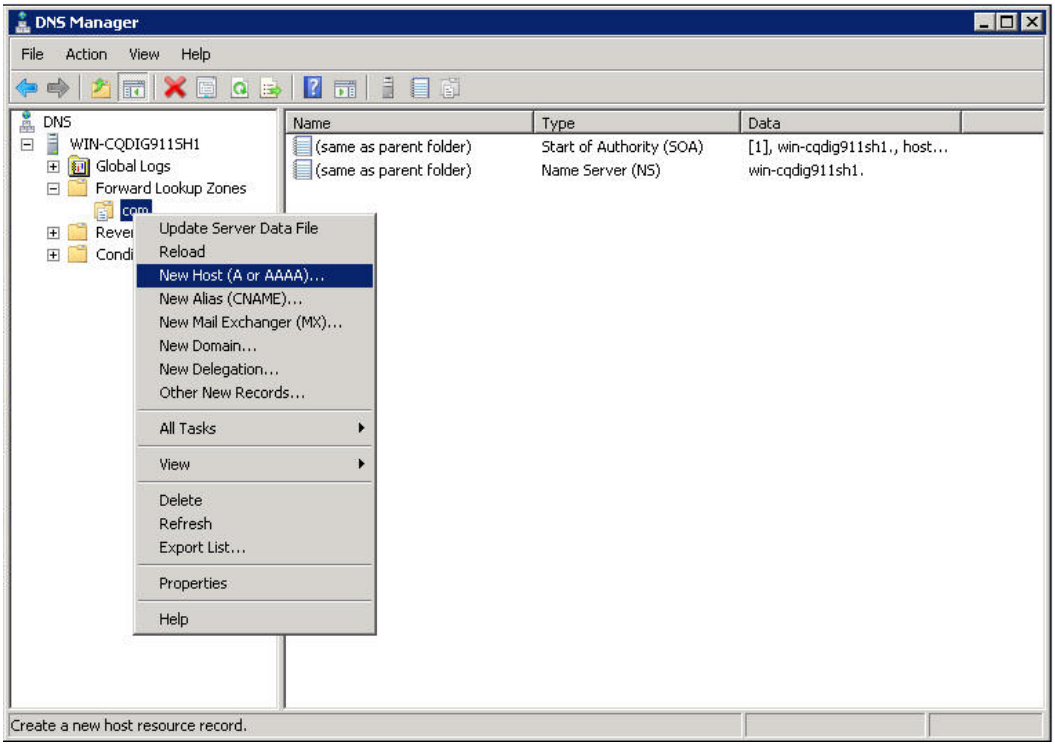
b. Right-click **Forward Lookup Zones**, select **New Zone**, and then follow the wizard to create a new zone named **com**.

Figure 6 Creating a zone



a. On the DNS server configuration page, right-click zone **com** and select **New Host**.

Figure 7 Adding a host

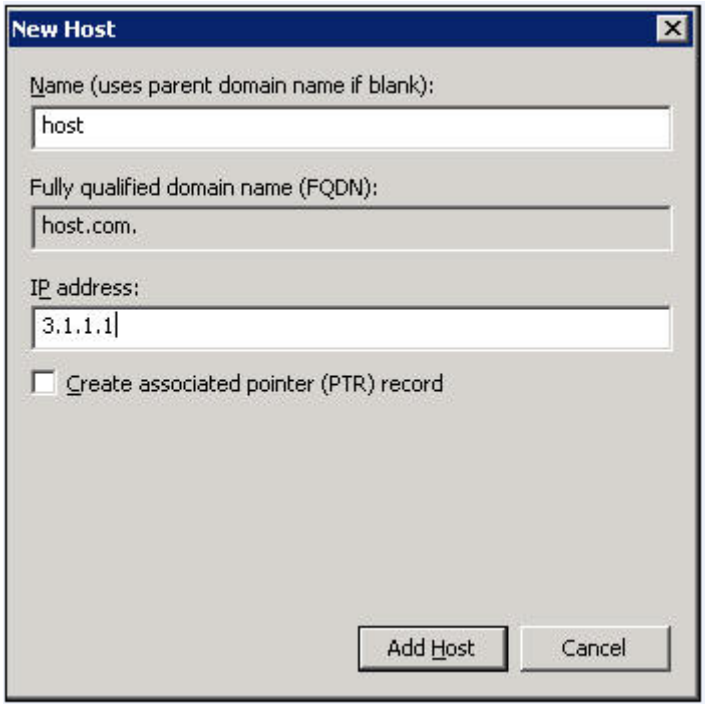


a. On the page that appears, enter host name **host** and IP address **3.1.1.1**.

b. Click **Add Host**.

The mapping between the IP address and host name is created.

Figure 8 Adding a mapping between domain name and IP address



2. Configure the DNS client:

Specify the DNS server 2.1.1.2.

```
<Sysname> system-view
[Sysname] dns server 2.1.1.2
```

Specify **com** as the name suffix.

```
[Sysname] dns domain com
```

Verifying the configuration

Verify that the device can use the dynamic domain name resolution to resolve domain name **host.com** into IP address 3.1.1.1.

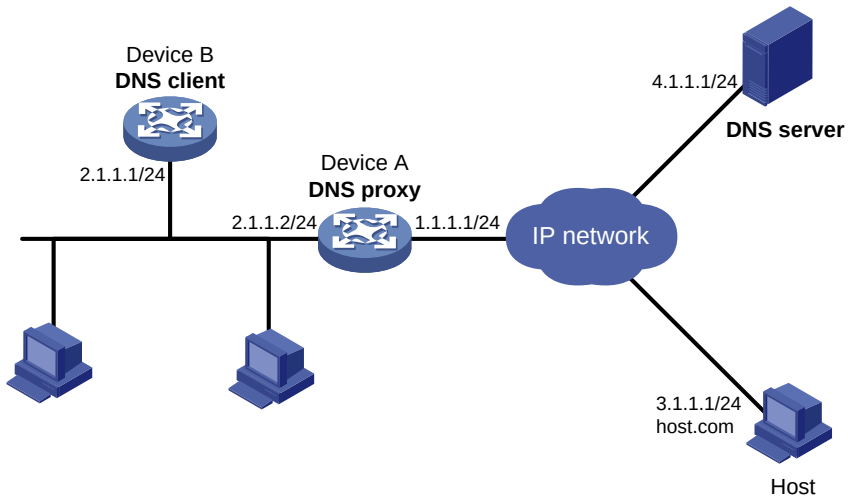
```
[Sysname] ping host
Ping host.com (3.1.1.1): 56 data bytes, press CTRL_C to break
56 bytes from 3.1.1.1: icmp_seq=0 ttl=255 time=1.000 ms
56 bytes from 3.1.1.1: icmp_seq=1 ttl=255 time=1.000 ms
56 bytes from 3.1.1.1: icmp_seq=2 ttl=255 time=1.000 ms
56 bytes from 3.1.1.1: icmp_seq=3 ttl=255 time=1.000 ms
56 bytes from 3.1.1.1: icmp_seq=4 ttl=255 time=2.000 ms

--- Ping statistics for host ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 1.000/1.200/2.000/0.400 ms
```

Example: Configuring DNS proxy

Network configuration

As shown in Figure 9, configure Device A as the DNS proxy to forward DNS packets between the DNS client (Device B) and the DNS server at 4.1.1.1.

Figure 9 Network diagram



Procedure

Before performing the following configuration, make sure that:

- Device A, the DNS server, and the host can reach each other.
- The IP addresses of the interfaces are configured as shown in Figure 9.

1. Configure the DNS server:

The configuration might vary by DNS server. When a PC running Windows Server 2008 R2 acts as the DNS server, see "Example: Configuring dynamic domain name resolution" for configuration information.

2. Configure the DNS proxy:

   # Specify the DNS server 4.1.1.1.

   ```
   <DeviceA> system-view
   [DeviceA] dns server 4.1.1.1
   ```

   # Enable DNS proxy.

   ```
   [DeviceA] dns proxy enable
   ```

3. Configure the DNS client:

   ```
   <DeviceB> system-view
   ```

   # Specify the DNS server 2.1.1.2.

   ```
   [DeviceB] dns server 2.1.1.2
   ```

Verifying the configuration

Verify that DNS proxy on Device A functions.

```
[DeviceB] ping host.com
Ping host.com (3.1.1.1): 56 data bytes, press CTRL_C to break
56 bytes from 3.1.1.1: icmp_seq=0 ttl=255 time=1.000 ms
56 bytes from 3.1.1.1: icmp_seq=1 ttl=255 time=1.000 ms
56 bytes from 3.1.1.1: icmp_seq=2 ttl=255 time=1.000 ms
56 bytes from 3.1.1.1: icmp_seq=3 ttl=255 time=1.000 ms
56 bytes from 3.1.1.1: icmp_seq=4 ttl=255 time=2.000 ms

--- Ping statistics for host.com ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 1.000/1.200/2.000/0.400 ms
```

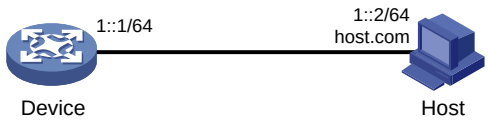
IPv6 DNS configuration examples

Example: Configuring static domain name resolution

Network configuration

As shown in Figure 10, the host at 1::2 is named **host.com**. Configure static IPv6 DNS on the device so that the device can use the easy-to-remember domain name rather than the IPv6 address to access the host.

Figure 10 Network diagram



Procedure

Configure a mapping between host name **host.com** and IPv6 address 1::2.

```
<Device> system-view
[Device] ipv6 host host.com 1::2
```

Verify that the device can use static domain name resolution to resolve domain name **host.com** into IPv6 address 1::2.

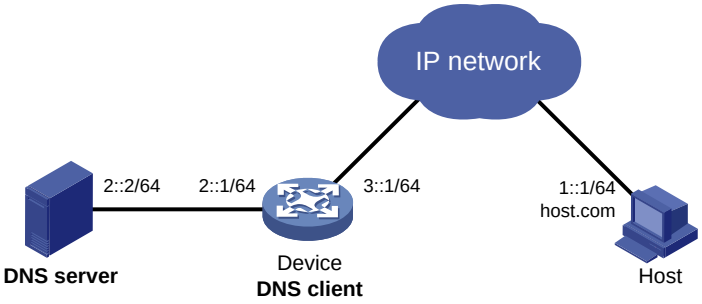
```
[Sysname] ping ipv6 host.com
Ping6(56 data bytes) 1::1 --> 1::2, press CTRL_C to break
56 bytes from 1::2, icmp_seq=0 hlim=128 time=1.000 ms
56 bytes from 1::2, icmp_seq=1 hlim=128 time=0.000 ms
56 bytes from 1::2, icmp_seq=2 hlim=128 time=1.000 ms
56 bytes from 1::2, icmp_seq=3 hlim=128 time=1.000 ms
56 bytes from 1::2, icmp_seq=4 hlim=128 time=0.000 ms

--- Ping6 statistics for host.com ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/0.600/1.000/0.490 ms
```

Example: Configuring dynamic domain name resolution

Network configuration

As shown in Figure 11, configure the DNS server to store the mapping between the host's domain name **host** and IPv6 address 1::1/64 in the **com** domain. Configure dynamic IPv6 DNS and DNS suffix **com** on the device so that the device can use domain name **host** to access the host.

Figure 11 Network diagram



Procedure

Before performing the following configuration, make sure that:

- The device and the host can reach each other.
- The IPv6 addresses of the interfaces are configured as shown in Figure 11.

1. Configure the DNS server:

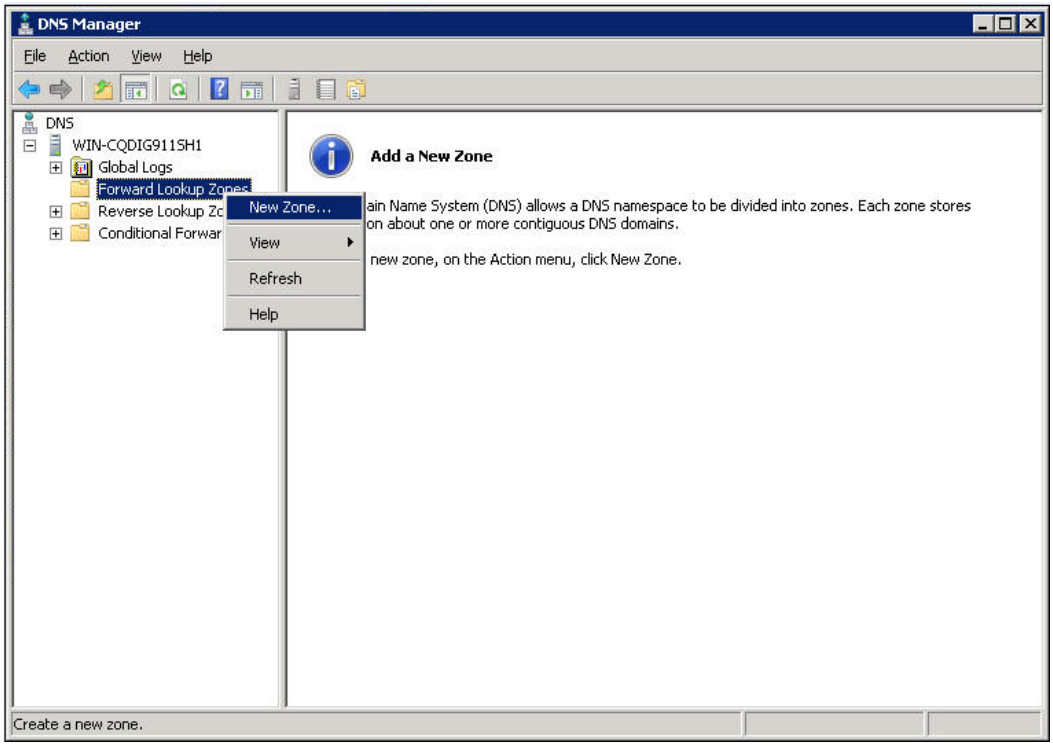
   The configuration might vary by DNS server. The following configuration is performed on a PC running Windows Server 2008 R2. Make sure that the DNS server supports IPv6 DNS so that the server can process IPv6 DNS packets and its interfaces can forward IPv6 packets.

   a. Select **Start** > **Programs** > **Administrative Tools** > **DNS**.

      The DNS server configuration page appears, as shown in Figure 12.

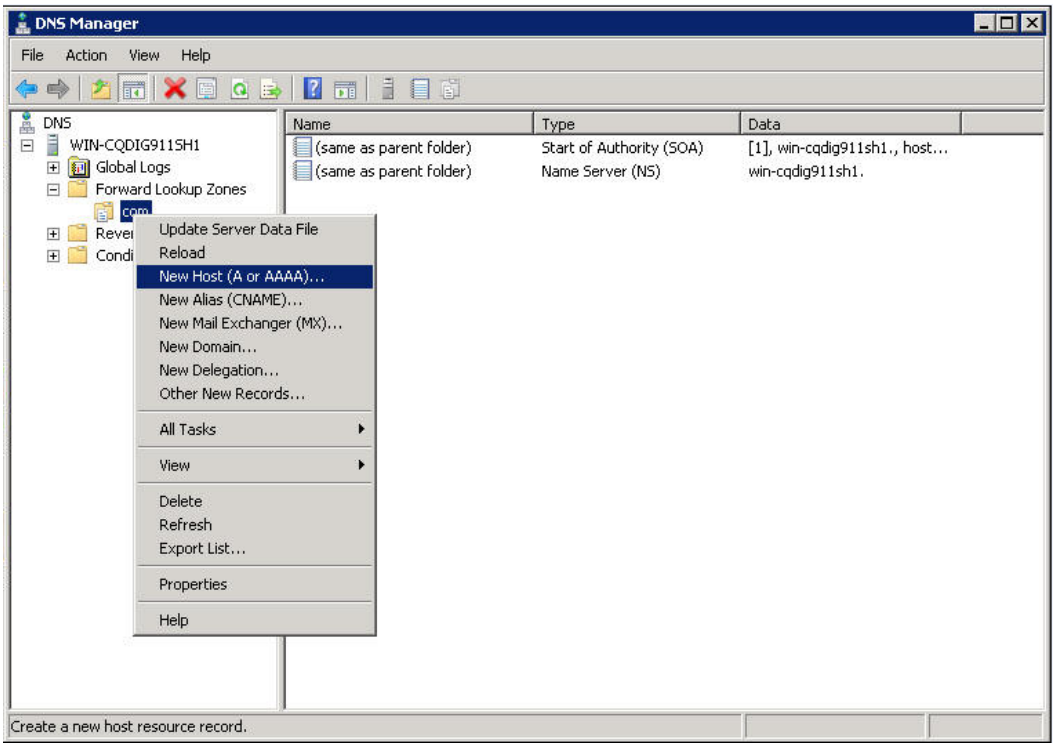
   b. Right-click **Forward Lookup Zones**, select **New Zone**, and then follow the wizard to create a new zone named **com**.

Figure 12 Creating a zone



a. On the DNS server configuration page, right-click zone **com** and select **New Host**.

Figure 13 Adding a host

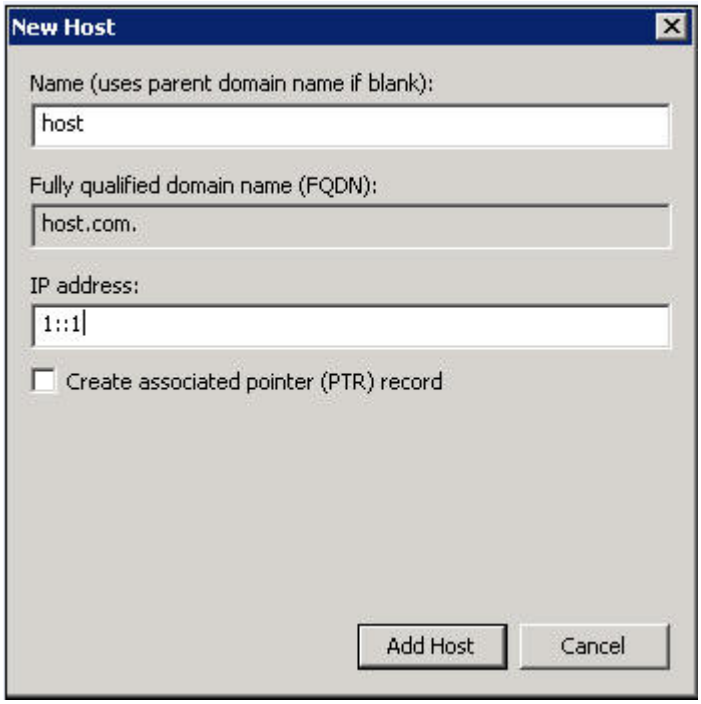


a. On the page that appears, enter host name **host** and IPv6 address **1::1**.

b. Click **Add Host**.

The mapping between the IPv6 address and host name is created.

Figure 14 Adding a mapping between domain name and IPv6 address



2. Configure the DNS client:

   # Specify the DNS server 2::2.

   ```
   <Device> system-view
   [Device] ipv6 dns server 2::2
   ```

   # Configure **com** as the DNS suffix.

   ```
   [Device] dns domain com
   ```

Verifying the configuration

Verify that the device can use the dynamic domain name resolution to resolve the domain name **host.com** into the IP address 1::1.

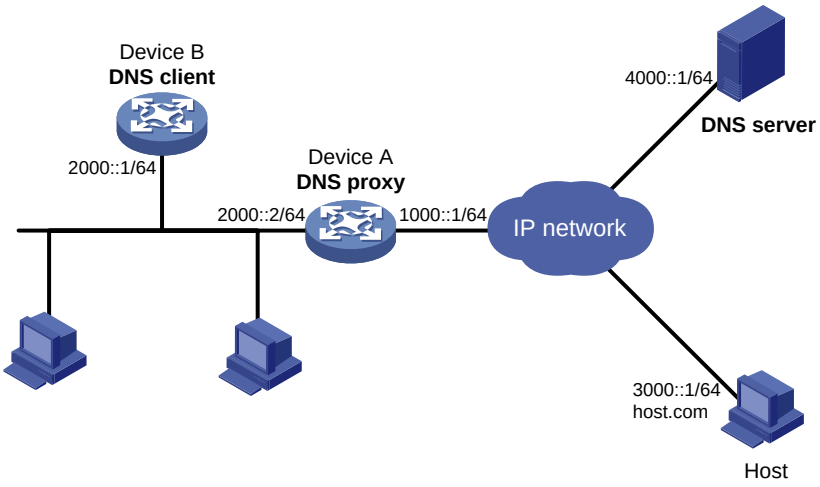
```
[Device] ping ipv6 host
Ping6(56 data bytes) 3::1 --> 1::1, press CTRL_C to break
56 bytes from 1::1, icmp_seq=0 hlim=128 time=1.000 ms
56 bytes from 1::1, icmp_seq=1 hlim=128 time=0.000 ms
56 bytes from 1::1, icmp_seq=2 hlim=128 time=1.000 ms
56 bytes from 1::1, icmp_seq=3 hlim=128 time=1.000 ms
56 bytes from 1::1, icmp_seq=4 hlim=128 time=0.000 ms

--- Ping6 statistics for host ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/0.600/1.000/0.490 ms
```

Example: Configuring DNS proxy

Network configuration

As shown in Figure 15, configure Device A as the DNS proxy to forward DNS packets between the DNS client (Device B) and the DNS server at 4000::1.

Figure 15 Network diagram



Procedure

Before performing the following configuration, make sure that:

- Device A, the DNS server, and the host are reachable to each other.
- The IPv6 addresses of the interfaces are configured as shown in Figure 15.

1. Configure the DNS server:

   This configuration might vary by DNS server. When a PC running Windows Server 2008 R2 acts as the DNS server, see "Example: Configuring dynamic domain name resolution" for configuration information.

2. Configure the DNS proxy:

   # Specify the DNS server 4000::1.

   ```
   <DeviceA> system-view
   [DeviceA] ipv6 dns server 4000::1
   ```

   # Enable DNS proxy.

   ```
   [DeviceA] dns proxy enable
   ```

3. Configure the DNS client:

   # Specify the DNS server 2000::2.

   ```
   <DeviceB> system-view
   [DeviceB] ipv6 dns server 2000::2
   ```

Verifying the configuration

Verify that DNS proxy on Device A functions.

```
[DeviceB] ping host.com
Ping6(56 data bytes) 2000::1 --> 3000::1, press CTRL_C to break
56 bytes from 3000::1, icmp_seq=0 hlim=128 time=1.000 ms
56 bytes from 3000::1, icmp_seq=1 hlim=128 time=0.000 ms
56 bytes from 3000::1, icmp_seq=2 hlim=128 time=1.000 ms
56 bytes from 3000::1, icmp_seq=3 hlim=128 time=1.000 ms
56 bytes from 3000::1, icmp_seq=4 hlim=128 time=0.000 ms

--- Ping6 statistics for host.com ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/0.600/1.000/0.490 ms
```

Troubleshooting DNS configuration

Failure to resolve IPv4 addresses

Symptom

After enabling dynamic domain name resolution, the user cannot get the correct IP address.

Solution

To resolve the problem:

1. Use the **`display dns host ip`** command to verify that the specified domain name is in the cache.
2. If the specified domain name does not exist, check that the DNS client can communicate with the DNS server.
3. If the specified domain name is in the cache, but the IP address is incorrect, check that the DNS client has the correct IP address of the DNS server.
4. Verify that the mapping between the domain name and IP address is correct on the DNS server.

Failure to resolve IPv6 addresses

Symptom

After enabling dynamic domain name resolution, the user cannot get the correct IPv6 address.

Solution

To resolve the problem:

1. Use the **`display dns host ipv6`** command to verify that the specified domain name is in the cache.
2. If the specified domain name does not exist, check that dynamic domain name resolution is enabled, and that the DNS client can communicate with the DNS server.
3. If the specified domain name is in the cache, but the IPv6 address is incorrect, check that the DNS client has the correct IPv6 address of the DNS server.
4. Verify that the mapping between the domain name and IPv6 address is correct on the DNS server.

Configuring DDNS

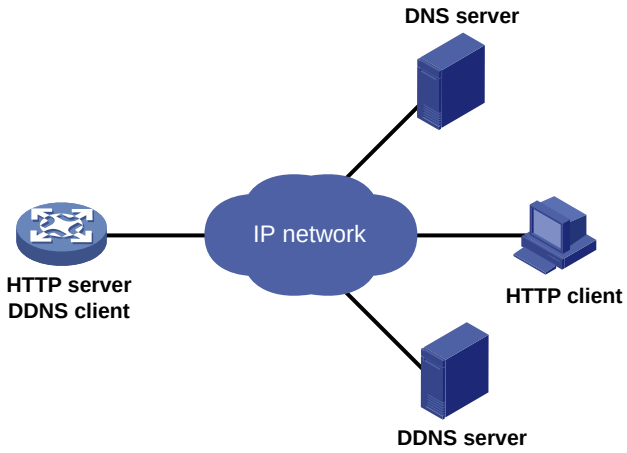
About DDNS

DNS provides only the static mappings between domain names and IP addresses. When the IP address of a node changes, your access to the node fails.

Dynamic Domain Name System (DDNS) can dynamically update the mappings between domain names and IP addresses for DNS servers.

Figure 16 shows the typical DDNS application.

Figure 16 DDNS application



DDNS works on the client-server model.

- **DDNS client**—A device that needs to update the mapping between its domain name and IP address dynamically on the DNS server when its IP address changes. An Internet user typically accesses an application layer server such as an HTTP server or an FTP server by using the server's domain name. When its IP address changes, the application layer server runs as a DDNS client. It sends a request to the DDNS server for updating the mapping between its domain name and its IP address.
- **DDNS server**—Informs the DNS server of latest mappings. When receiving the mapping update request from a DDNS client, the DDNS server tells the DNS server to re-map the domain name and the IP address of the DDNS client. Therefore, the Internet users can use the same domain name to access the DDNS client even if the IP address of the DDNS client has changed.

The device can function as a DDNS client to update the domain name-IP address mappings on the DNS servers through DDNS servers such as www.3322.org and PeanutHull.

NOTE:

The DDNS update process does not have a unified standard but varies by DDNS server that the DDNS client contacts.

Restrictions and guidelines: DDNS configuration

DDNS is supported by only IPv4 DNS. It is used to update the mappings between domain names and IPv4 addresses.

DDNS client tasks at a glance

To configure a DDNS client, perform the following tasks:

1. Configuring a DDNS policy
2. Applying the DDNS policy to an interface
3. (Optional.) Setting the DSCP value for outgoing DDNS packets

Configuring a DDNS policy

About DDNS policy

A DDNS policy contains the DDNS server address, port number, login ID, password, time interval, associated SSL client policy, and update time interval. After creating a DDNS policy, you can apply it to multiple interfaces to simplify DDNS configuration.

Restrictions and guidelines

The URL address for update requests varies by DDNS server.

Table 1 Common URL addresses

DDNS server	URL address for DDNS update requests
www.3322.org	http://members.3322.org/dyndns/update?system=dyndns&hostname=<h>&myip=<a>
DYNDNS	http://members.dyndns.org/nic/update?system=dyndns&hostname=<h>&myip=<a>
DYNS	http://www.dyns.cx/postscript.php?host=<h>&ip=<a>
ZONEEDIT	http://dynamic.zoneedit.com/auth/dynamic.html?host=<h>&dnsto=<a>
TZO	http://cgi.tzo.com/webclient/signedon.html?TZOName=<h>IPAddress=<a>
EASYDNS	http://members.easydns.com/dyn/ez-ipupdate.php?action=edit&myip=<a>&host_id=<h>
HEIPV6TB	http://dyn.dns.he.net/nic/update?hostname=<h>&myip=<a>
CHANGE-IP	http://nic.changeip.com/nic/update?hostname=<h>&offline=1
NO-IP	http://dynupdate.no-ip.com/nic/update?hostname=<h>&myip=<a>
DHS	http://members.dhs.org/nic/hosts?domain=dyn.dhs.org&hostname=<h>&hostscmd=edit&hostscmdstage=2&type=1&ip=<a>
HP	https://server-name/nic/update?group=group-name&myip=<a>
ODS	ods://update.ods.org
GNUDIP	gnudip://*server-name*
PeanutHull	Select the URL according to your network situation: • oray://phservice2.oray.net • oray://phddns60.oray.net • oray://client.oray.net • oray://ph031.oray.net

Identify the DDNS server type in your network and follow the following restrictions and guidelines to set an appropriate URL address:

- The URL address for an update request can start with:
  - **http://**—The HTTP-based DDNS server.

  - **https://**—The HTTPS-based DDNS server.
  - **ods://**—The TCP-based ODS server.
  - **gnudip://**—The TCP-based GNUDIP server.
  - **oray://**—The TCP-based PeanutHull DDNS server.
- HP and GNUDIP are common DDNS update protocols. The *server-name* argument is the domain name or IP address of the service provider's server using one of the update protocols.
- The port number in the URL address is optional. If no port is specified, the system uses the default port numbers: port 80 for HTTP, port 443 for HTTPS, and port 6060 for PeanutHull DDNS server.
- The <h> value can be automatically filled with an FQDN if it is specified in the command for applying a DDNS policy to an interface. The <a> value is automatically filled with the primary IP address of the interface to which the DDNS policy is applied. For more information about applying DDNS policies, see "Applying the DDNS policy to an interface."
- You can also manually specify an FQDN and an IP address for the <h> and <a> fields. In this case, the FQDN specified at the CLI does not take effect. As a best practice, do not manually change the <h> and <a> because your configuration might be incorrect.
- No FQDN or IP address can be specified in the URL address for update requests sent to the PeanutHull DDNS server. You can specify the FQDN when applying the DDNS policy to an interface. The IP address is the primary IP address of the interface to which the DDNS policy is applied.

Prerequisites

Visit the website of a DDNS service provider, register an account, and apply for a domain name for the DDNS client. When the DDNS client updates the mapping between the domain name and the IP address through the DDNS server, the DDNS server checks the following:

- Whether the account information is correct.
- Whether the domain name to be updated belongs to the account.

Procedure

1. Enter system view.

   system-view
2. Create a DDNS policy and enter its view.

   ddns policy *policy-name*
3. Specify a URL address for DDNS update requests.

   url *request-url*

   By default, no URL address is specified for DDNS update requests.

   The URL address cannot contain a username or password. To configure them, use the **username** command and the **password** command.
4. Specify the username for logging in to the DDNS server.

   username *username*

   By default, no username is specified.
5. Specify the password for logging in to the DDNS server.

   password { **cipher** | **simple** } *string*

   By default, no password is specified.
6. (Optional.) Specify the parameter transmission method for sending DDNS update requests to HTTP/HTTPS-based DDNS servers.

   method { **http-get** | **http-post** }

   By default, the **http-get method** is used.

This step is effective for communicating with HTTP/HTTPS-based DDNS servers.

Specify the **http-post** keyword for DDNS update with a DHS server.

7. (Optional.) Associate an SSL client policy with the DDNS policy.

   ssl-client-policy *policy-name*

   By default, no SSL client policy is associated with the DDNS policy.

   This step is only effective and a must for HTTP-based DDNS update requests. For SSL client policy configuration, see *Security Configuration Guide*.

8. (Optional.) Specify the interval for sending update requests.

   interval *days* [*hours* [*minutes*]]

   By default, the time interval is one hour.

Applying the DDNS policy to an interface

About DDNS policy application to an interface

After you apply the DDNS policy to an interface and specify the FQDN for update, the DDNS client can send requests to the DDNS server. The requests are to update the mapping between the domain name and the primary IP address of the interface.

Restrictions and guidelines

- The **fqdn** *domain-name* option is a must for all DDNS servers except the PeanutHull DDNS server.
- The **fqdn** *domain-name* option is optional for PeanutHull DDNS server. If no FQDN is specified, the DDNS server updates all domain names for the DDNS client account. If an FQDN is specified, the DDNS server updates only the mapping between the specified FQDN and the primary IP address.

Prerequisites

Before you apply a DDNS policy to an interface, complete the following tasks:

- Specify the primary IP address of the interface and make sure the DDNS server and the interface can reach each other.
- Configure static or dynamic domain name resolution to translate the domain name of the DDNS server into the IPv4 address. For more information, see "Configuring the DNS client."

Procedure

1. Enter system view.

   system-view

2. Enter interface view.

   interface *interface-type interface-number*

3. Apply the DDNS policy to the interface to update the mapping between the specified FQDN and the primary IP address of the interface, and enable DDNS update.

   ddns apply policy *policy-name* [**fqdn** *domain-name*]

   By default, no DDNS policy is applied to the interface, no FQDN is specified for update, and DDNS update is disabled.

   An FQDN, including a host name and a domain name, is the only identifier for a network node and can be resolved as an IP address.

Setting the DSCP value for outgoing DDNS packets

About the DSCP value for outgoing DDNS packets

The DSCP value of a packet specifies the priority level of the packet and affects the transmission priority of the packet. A bigger DSCP value represents a higher priority.

Procedure

1. Enter system view.
   system-view
2. Set the DSCP value for outgoing DDNS packets.
   ddns dscp *dscp-value*
   By default, the DSCP value for outgoing DDNS packets is 0.

Display and maintenance commands for DDNS

Execute **display** commands in any view.

Task	Command
Display DDNS policy information.	**display ddns policy** [*policy-name*]

DDNS configuration examples

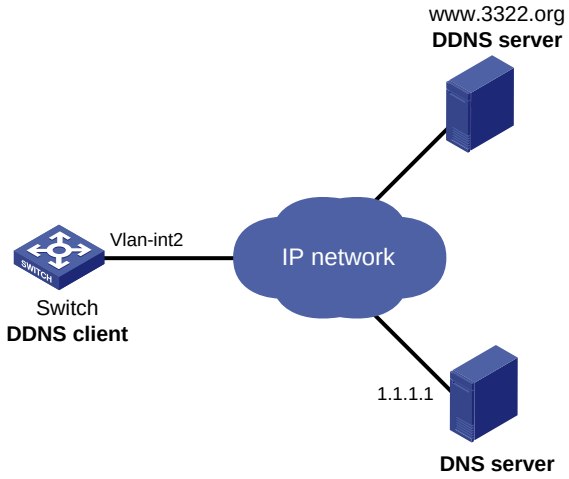
Example: Configuring DDNS with www.3322.org

Network configuration

As shown in Figure 17, the switch is a Web server with domain name **whatever.3322.org** and uses an IP address dynamically obtained through DHCP. To make sure the switch can always provide Web services at **whatever.3322.org** when its IP address changes, perform the following tasks on the switch:

- Configure a DDNS policy to update the switch's domain name-to-IP address mapping on the DDNS server. The DDNS server then updates the mapping on the DNS server.
- Specify the IP address of the DNS server so that the switch can access the DDNS server through domain name.

Figure 17 Network diagram



Procedure

Before configuring DDNS on the switch, perform the following tasks:

- Register with username **steven** and password **nevets** at http://www.3322.org/.
- Configure a DDNS policy to update the mapping between the switch's FQDN and IP address.
- Make sure the devices can reach each other.

Create a DDNS policy named **3322.org**, and enter its view.

```
<Switch> system-view
[Switch] ddns policy 3322.org
```

Specify the URL address, username, and password for DDNS update requests.

```
[Switch-ddns-policy-3322.org] url
http://members.3322.org/dyndns/update?system=dyndns&hostname=<h>&myip=<a>
[Switch-ddns-policy-3322.org] username steven
[Switch-ddns-policy-3322.org] password simple nevets
```

Set the interval to 15 minutes for sending DDNS update requests.

```
[Switch-ddns-policy-3322.org] interval 0 0 15
[Switch-ddns-policy-3322.org] quit
```

Specify the IP address of the DNS server as 1.1.1.1.

```
[Switch] dns server 1.1.1.1
```

Apply DDNS policy **3322.org** to VLAN-interface 2 to enable DDNS update. The mapping between domain name **whatever.3322.org** and the primary IP address of VLAN-interface 2 will be dynamically updated.

```
[Switch] interface vlan-interface 2
[Switch-Vlan-interface2] ddns apply policy 3322.org fqdn whatever.3322.org
```

Verifying the configuration

Verify that the switch can update its domain name-IP mapping through the DDNS provider **www.3322.org** when its IP address changes. The Internet users can resolve the correct IP address through the domain name **whatever.3322.org** to access the Web service.

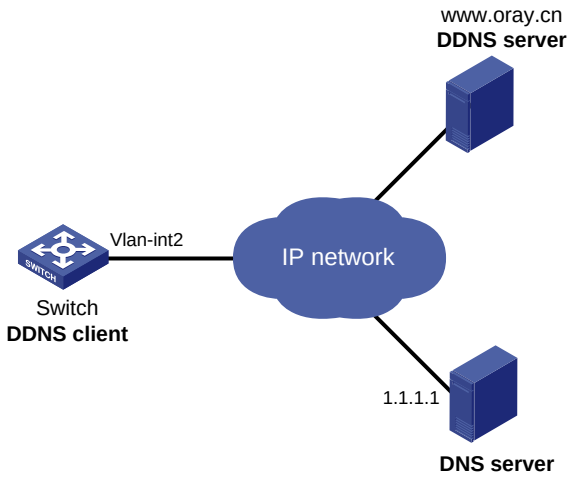
Example: Configuring DDNS with PeanutHull server

Network configuration

As shown in Figure 18, the switch is a Web server with domain name **whatever.gicp.cn** and uses an IP address dynamically obtained through DHCP. To make sure the switch can always provide Web services at **whatever.gicp.cn** when its IP address changes, perform the following tasks on the switch:

- Configure a DDNS policy to update the switch's domain name-to-IP address mapping on the DDNS server. The DDNS server then updates the mapping on the DNS server.
- Specify the IP address of the DNS server so that the switch can access the DDNS server through domain name.

Figure 18 Network diagram



Procedure

Before configuring DDNS on the switch, perform the following tasks:

- Register with username **steven** and password **nevets** at http://www.oray.cn/.
- Configure a DDNS policy to update the mapping between the switch's FQDN and IP address.
- Make sure the devices can reach each other.

Create a DDNS policy named **oray.cn** and enter its view.

```
<Switch> system-view
[Switch] ddns policy oray.cn
```

Specify the URL address, username, and password for DDNS update requests.

```
[Switch-ddns-policy-oray.cn] url oray://phservice2.oray.net
[Switch-ddns-policy-oray.cn] username steven
[Switch-ddns-policy-oray.cn] password simple nevets
```

Set the DDNS update request interval to 12 minutes.

```
[Switch-ddns-policy-oray.cn] interval 0 0 12
[Switch-ddns-policy-oray.cn] quit
```

Specify the IP address of the DNS server as 1.1.1.1.

```
[Switch] dns server 1.1.1.1
```

Apply DDNS policy **oray.cn** to VLAN-interface 2 to enable DDNS update. The mapping between domain name **whatever.gicp.cn** and the primary IP address of VLAN-interface 2 will be dynamically updated.

```
[Switch] interface vlan-interface 2
[Switch-Vlan-interface2] ddns apply policy oray.cn fqdn whatever.gicp.cn
```

Verifying the configuration

Verify that the switch can update its domain name-IP mapping through the Peanuthull DDNS provider when its IP address changes. The Internet users can resolve the correct IP address through the domain name **whatever.gicp.cn** to access the Web service.