



H3C S5120-EI Switch Series Layer 3 - IP Services Configuration Guide

Hangzhou H3C Technologies Co., Ltd.
<http://www.h3c.com>

Software version: Release 2220
Document version: 6W100-20130810

Copyright © 2013, Hangzhou H3C Technologies Co., Ltd. and its licensors

All rights reserved

No part of this manual may be reproduced or transmitted in any form or by any means without prior written consent of Hangzhou H3C Technologies Co., Ltd.

Trademarks

H3C, **H3C**, H3CS, H3CIE, H3CNE, Aolynk, , H³Care, , IRF, NetPilot, Netflow, SecEngine, SecPath, SecCenter, SecBlade, Comware, ITCMM and HUASAN are trademarks of Hangzhou H3C Technologies Co., Ltd.

All other trademarks that may be mentioned in this manual are the property of their respective owners

Notice

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute the warranty of any kind, express or implied.

Preface

The H3C S5120-EI documentation set includes 10 configuration guides, which describe the software features for the H3C S5120-EI Switch Series Release 2220, and guide you through the software configuration procedures. These configuration guides also provide configuration examples to help you apply software features to different network scenarios.

The *Layer 3—IP Services Configuration Guide* describes how to configure IP addressing, DHCP, IP performance optimization, ARP, DNS, UDP helper, IRDP, IPv6 basics, and DHCPv6.

This preface includes:

- [Audience](#)
- [Added and modified features](#)
- [Conventions](#)
- [About the S5120-EI documentation set](#)
- [Obtaining documentation](#)
- [Technical support](#)
- [Documentation feedback](#)

Audience

This documentation is intended for:

- Network planners
- Field technical support and servicing engineers
- Network administrators working with the S5120-EI series

Added and modified features

Compared to Release 2210, Release 2220 adds and modifies the following features:

Configuration guide	Added and modified features
ARP	Added feature: enabling IP conflict notification
IP addressing	Added feature: supporting a subnet mask of 31 bits long.
DHCP	Added features: <ul style="list-style-type: none">• Enabling client offline detection• Setting the DSCP value for DHCP protocol packets sent by the DHCP server, DHCP relay agent, and DHCP client.
DNS	Added features: <ul style="list-style-type: none">• Setting the DSCP value for DNS protocol packets transmitted.• Specifying the source interface for DNS packets.
IRDP	N/A
IP performance optimization	N/A

Configuration guide	Added and modified features
UDP helper	N/A
IPv6 basics	N/A
DHCPv6	Added features: <ul style="list-style-type: none"> Setting the DSCP value for DHCPv6 protocol packets sent by the DHCPv6 server DHCPv6 relay agent, and DHCPv6 client. Configuring DHCPv6 snooping to support Option 18 and Option 37.
IPv6 DNS	Added feature: setting the DSCP value for IPv6 DNS protocol packets.

Conventions

This section describes the conventions used in this documentation set.





Command conventions

Convention	Description
Boldface	Bold text represents commands and keywords that you enter literally as shown.
<i>Italic</i>	<i>Italic</i> text represents arguments that you replace with actual values.
[]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x y ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x y ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x y ... } *	Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select at least one.
[x y ...] *	Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.




GUI conventions

Convention	Description
Boldface	Window names, button names, field names, and menu items are in Boldface. For example, the New User window appears; click OK .
>	Multi-level menus are separated by angle brackets. For example, File > Create > Folder .
Convention	Description
< >	Button names are inside angle brackets. For example, click <OK>.
[]	Window names, menu items, data table and field names are inside square brackets. For example, pop up the [New User] window.
/	Multi-level menus are separated by forward slashes. For example, [File/Create/Folder].

Symbols

Convention	Description
 WARNING	An alert that calls attention to important information that if not understood or followed can result in personal injury.
 CAUTION	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
 IMPORTANT	An alert that calls attention to essential information.
NOTE	An alert that contains additional or supplementary information.
 TIP	An alert that provides helpful information.

Network topology icons

	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.

Port numbering in examples

The port numbers in this document are for illustration only and might be unavailable on your device.

About the S5120-EI documentation set

The H3C S5120-EI documentation set includes:

Category	Documents	Purposes
Product description and specifications	Marketing brochure	Describe product specifications and benefits.
	Technology white papers	Provide an in-depth description of software features and technologies.
	Compliance and safety manual CE DOCs	Provide regulatory information and the safety instructions that must be followed during installation.
Hardware specifications and installation	Quick start	Guides you through initial installation and setup procedures to help you quickly set up your device.
	Installation guide	Provides a complete guide to switch installation and specifications.
	RPS Ordering Information for H3C Low-End Ethernet Switches	Helps you order RPSs for switches that can work with an RPS.
	User manuals for RPSs	Describe the specifications, installation, and replacement of RPSs.

Category	Documents	Purposes
	User manuals for interface cards	Describe the specifications, installation, and replacement of expansion interface cards.
	H3C Low End Series Ethernet Switches Pluggable Modules Manual	Describes the specifications of pluggable transceiver modules.
	Pluggable SFP[SFP+][XFP] Transceiver Modules Installation Guide	Describe the installation, and replacement of SFP/SFP+/XFP transceiver modules.
Software configuration	Configuration guides	Describe software features and configuration procedures.
	Command references	Provide a quick reference to all available commands.
Operations and maintenance	Release notes	Provide information about the product release, including the version history, hardware and software compatibility matrix, version upgrade information, technical support information, and software upgrading.

Obtaining documentation

You can access the most up-to-date H3C product documentation on the World Wide Web at <http://www.h3c.com>.

Click the links on the top navigation bar to obtain different categories of product documentation:

[\[Technical Support & Documents > Technical Documents\]](#) – Provides hardware installation, software upgrading, and software feature configuration and maintenance documentation.

[\[Products & Solutions\]](#) – Provides information about products and technologies, as well as solutions.

[\[Technical Support & Documents > Software Download\]](#) – Provides the documentation released with the software version.

Technical support

service@h3c.com

<http://www.h3c.com>

Documentation feedback

You can e-mail your comments about product documentation to info@h3c.com.

We appreciate your comments.

Contents

Configuring ARP.....	1
Overview.....	1
ARP message format.....	1
ARP operation.....	1
ARP table.....	2
Configuring a static ARP entry.....	3
Configuring the maximum number of dynamic ARP entries for an interface.....	4
Setting the aging timer for dynamic ARP entries.....	4
Enabling dynamic ARP entry check.....	4
Configuring ARP quick update.....	5
Configuring multicast ARP.....	5
Displaying and maintaining ARP.....	6
ARP configuration examples.....	7
Static ARP entry configuration example.....	7
Multicast ARP configuration example.....	8
Configuring gratuitous ARP.....	10
Overview.....	10
Gratuitous ARP packet learning.....	10
Periodic sending of gratuitous ARP packets.....	10
Configuration guidelines.....	10
Configuration procedure.....	11
Enabling IP conflict notification.....	11
Configuring proxy ARP.....	12
Overview.....	12
Common proxy ARP.....	12
Local proxy ARP.....	12
Enabling common proxy ARP.....	13
Enabling local proxy ARP.....	13
Displaying and maintaining proxy ARP.....	13
Proxy ARP configuration examples.....	14
Common proxy ARP configuration example.....	14
Local proxy ARP configuration example in case of port isolation.....	15
Local proxy ARP configuration example in isolate-user-VLAN.....	16
Configuring ARP snooping.....	18
Overview.....	18
Configuration procedure.....	18
Displaying and maintaining ARP snooping.....	18
Configuring IP addressing.....	19
Overview.....	19
IP address classes.....	19
Special IP addresses.....	20
Subnetting and masking.....	20
Assigning an IP address to an interface.....	21
Configuration guidelines.....	21
Configuration procedure.....	21
Configuration example.....	21
Displaying and maintaining IP addressing.....	23

DHCP overview	24
DHCP address allocation	24
Dynamic IP address allocation process	25
IP address lease extension	25
DHCP message format	26
DHCP options	27
Common DHCP options	27
Custom options	27
Protocols and standards	31
Configuring DHCP server	33
Overview	33
DHCP address pool	33
IP address allocation sequence	34
DHCP server configuration task list	34
Configuring an address pool for the DHCP server	35
Configuration task list	35
Creating a DHCP address pool	35
Configuring address allocation mode for a common address pool	36
Configuring dynamic address allocation for an extended address pool	38
Configuring a domain name suffix for the client	38
Configuring DNS servers for the client	39
Configuring WINS servers and NetBIOS node type for the client	39
Configuring BIMS server information for the client	40
Configuring gateways for the client	40
Configuring Option 184 parameters for the client with voice service	40
Configuring the TFTP server and bootfile name for the client	41
Specifying a server's IP address for the DHCP client	42
Configuring self-defined DHCP options	42
Enabling DHCP	43
Enabling the DHCP server on an interface	43
Configuration guidelines	43
Configuration procedure	44
Applying an extended address pool on an interface	44
Configuring the DHCP server security functions	44
Configuration prerequisites	44
Enabling unauthorized DHCP server detection	44
Configuring IP address conflict detection	45
Enabling client offline detection	45
Enabling handling of Option 82	46
Configuration prerequisites	46
Enabling Option 82 handling	46
Specifying the threshold for sending trap messages	46
Configuration prerequisites	46
Configuration procedure	46
Setting the DSCP value for DHCP packets	47
Displaying and maintaining the DHCP server	47
DHCP server configuration examples	48
Static IP address assignment configuration example	48
Dynamic IP address assignment configuration example	49
Self-defined option configuration example	51
Troubleshooting DHCP server configuration	52
Symptom	52
Analysis	52
Solution	52

Configuring DHCP relay agent	53
Overview	53
Fundamentals	53
DHCP relay agent support for Option 82	54
DHCP relay agent configuration task list	54
Enabling DHCP	55
Enabling the DHCP relay agent on an interface	55
Correlating a DHCP server group with a relay agent interface	55
Configuration guidelines	55
Configuration procedure	56
Configuring the DHCP relay agent security functions	56
Configuring address check	56
Configuring periodic refresh of dynamic client entries	57
Enabling unauthorized DHCP server detection	57
Enabling DHCP starvation attack protection	58
Enabling offline detection	59
Configuring the DHCP relay agent to release an IP address	59
Configuring the DHCP relay agent to support Option 82	59
Configuration prerequisites	59
Configuration guidelines	60
Configuration procedure	60
Setting the DSCP value for DHCP packets	61
Displaying and maintaining the DHCP relay agent	61
DHCP relay agent configuration examples	62
DHCP relay agent configuration example	62
DHCP relay agent Option 82 support configuration example	63
Troubleshooting DHCP relay agent configuration	63
Symptom	63
Analysis	63
Solution	64
Configuring DHCP client	65
Configuration restrictions	65
Enabling the DHCP client on an interface	65
Setting the DSCP value for DHCP packets	66
Displaying and maintaining the DHCP client	66
DHCP client configuration example	66
Network requirements	66
Configuration procedure	67
Verifying the configuration	67
Configuring DHCP snooping	69
DHCP snooping functions	69
Ensuring that DHCP clients obtain IP addresses from authorized DHCP servers	69
Recording IP-to-MAC mappings of DHCP clients	69
Application environment of trusted ports	69
Configuring a trusted port connected to a DHCP server	69
Configuring trusted ports in a cascaded network	70
DHCP snooping support for Option 82	71
DHCP snooping configuration task list	72
Configuring DHCP snooping basic functions	72
Configuring DHCP snooping to support Option 82	73
Configuring DHCP snooping entries backup	75
Enabling DHCP starvation attack protection	75
Enabling DHCP-REQUEST message attack protection	76

Configuring DHCP packet rate limit.....	76
Displaying and maintaining DHCP snooping.....	77
DHCP snooping configuration examples.....	77
DHCP snooping configuration example.....	77
DHCP snooping Option 82 support configuration example.....	78
Configuring BOOTP client.....	80
Overview.....	80
BOOTP application.....	80
Obtaining an IP address dynamically.....	80
Protocols and standards.....	80
Configuration restrictions.....	80
Configuring an interface to dynamically obtain an IP address through BOOTP.....	81
Displaying and maintaining BOOTP client configuration.....	81
BOOTP client configuration example.....	81
Network requirements.....	81
Configuration procedure.....	81
Configuring IPv4 DNS.....	82
Overview.....	82
Static domain name resolution.....	82
Dynamic domain name resolution.....	82
DNS proxy.....	83
DNS spoofing.....	84
Configuring the IPv4 DNS client.....	85
Configuring static domain name resolution.....	85
Configuring dynamic domain name resolution.....	85
Configuring the DNS proxy.....	86
Configuring DNS spoofing.....	87
Setting the DSCP value for DNS packets.....	87
Specifying the source interface for DNS packets.....	87
Displaying and maintaining IPv4 DNS.....	88
Static domain name resolution configuration example.....	88
Network requirements.....	88
Configuration procedure.....	89
Dynamic domain name resolution configuration example.....	89
Network requirements.....	89
Configuration procedure.....	90
Verifying the configuration.....	92
DNS proxy configuration example.....	92
Network requirements.....	92
Configuration procedure.....	93
Verifying the configuration.....	93
Troubleshooting IPv4 DNS configuration.....	94
Symptom.....	94
Solution.....	94
Configuring IRDP.....	95
Overview.....	95
Background.....	95
Working mechanism.....	95
Concepts.....	96
Protocols and standards.....	96
Configuration procedure.....	96
IRDP configuration example.....	97
Network requirements.....	97

Configuration procedure	98
Verifying the configuration	99
Optimizing IP performance	100
Enabling receiving and forwarding of directed broadcasts to a directly connected network	100
Enabling receiving of directed broadcasts to a directly connected network	100
Enabling forwarding of directed broadcasts to a directly connected network	100
Configuration example	101
Configuring TCP attributes	101
Configuring TCP path MTU discovery	101
Configuring the TCP send/receive buffer size	102
Configuring TCP timers	102
Configuring ICMP to send error packets	103
Advantages of sending ICMP error packets	103
Disadvantages of sending ICMP error packets	104
Configuration procedure	104
Displaying and maintaining IP performance optimization	105
Configuring UDP helper	106
Overview	106
Configuration restrictions and guidelines	106
Configuration procedure	106
Displaying and maintaining UDP helper	107
UDP helper configuration example	107
Network requirements	107
Configuration procedure	107
Configuring IPv6 basics	109
Overview	109
IPv6 features	109
IPv6 addresses	110
IPv6 neighbor discovery protocol	113
IPv6 path MTU discovery	115
IPv6 transition technologies	116
Protocols and standards	116
IPv6 basics configuration task list	117
Configuring basic IPv6 functions	118
Enabling IPv6	118
Configuring an IPv6 global unicast address	118
Configuring an IPv6 link-local address	120
Configure an IPv6 anycast address	121
Configuring IPv6 ND	122
Configuring a static neighbor entry	122
Configuring the maximum number of neighbors dynamically learned	122
Setting the age timer for ND entries in stale state	123
Configuring parameters related to RA messages	123
Configuring the maximum number of attempts to send an NS message for DAD	125
Configuring ND snooping	126
Enabling ND proxy	128
Configuring path MTU discovery	130
Configuring a static path MTU for a specific IPv6 address	130
Configuring the aging time for dynamic path MTUs	130
Configuring IPv6 TCP properties	130
Configuring ICMPv6 packet sending	131
Configuring the maximum ICMPv6 error packets sent in an interval	131
Enabling replying to multicast echo requests	131

Enabling sending ICMPv6 time exceeded messages	132
Enabling sending ICMPv6 destination unreachable messages	132
Displaying and maintaining IPv6 basics configuration	133
IPv6 basics configuration example	134
Network requirements	134
Configuration procedure	134
Verifying the configuration	136
Troubleshooting IPv6 basics configuration	139
Symptom	139
Solution	139
DHCPv6 overview	140
Introduction to DHCPv6	140
DHCPv6 address/prefix assignment	140
Rapid assignment involving two messages	140
Assignment involving four messages	140
Address/prefix lease renewal	141
Configuring stateless DHCPv6	142
Operation	142
Protocols and standards	143
Configuring DHCPv6 server	144
Overview	144
Concepts	144
Prefix selection process	145
DHCPv6 server configuration task list	145
Enabling the DHCPv6 server	146
Creating a prefix pool	146
Configuring a DHCPv6 address pool	146
Configuration restrictions and guidelines	146
Configuration procedure	146
Applying the address pool to an interface	147
Setting the DSCP value for DHCPv6 packets	148
Displaying and maintaining the DHCPv6 server	148
DHCPv6 server configuration example	149
Network requirements	149
Configuration considerations	149
Configuration procedure	149
Verifying the configuration	150
Configuring DHCPv6 relay agent	152
Overview	152
DHCPv6 relay agent operation	152
Configuring the DHCPv6 relay agent	153
Configuration guidelines	153
Configuration procedure	153
Setting the DSCP value for DHCPv6 packets	154
Displaying and maintaining the DHCPv6 relay agent	154
DHCPv6 relay agent configuration example	154
Network requirements	154
Configuration procedure	155
Verifying the configuration	155
Configuring DHCPv6 client	157
Overview	157
Configuring the DHCPv6 client	157

Configuration prerequisites	157
Configuration guidelines	157
Configuration procedure	157
Setting the DSCP value for DHCPv6 packets	157
Displaying and maintaining the DHCPv6 client	158
Stateless DHCPv6 configuration example	158
Network requirements	158
Configuration procedure	158
Verifying the configuration	159
Configuring DHCPv6 snooping	161
Overview	161
Ensuring that DHCPv6 clients obtain IPv6 addresses from authorized DHCPv6 servers	161
Recording IP-to-MAC mappings of DHCPv6 clients	162
Enabling DHCPv6 snooping	162
Configuring a DHCPv6 snooping trusted port	162
Configuring the maximum number of DHCPv6 snooping entries an interface can learn	163
Configuring DHCPv6 snooping to support Option 18 and Option 37	163
Displaying and maintaining DHCPv6 snooping	164
DHCPv6 snooping configuration example	165
Network requirements	165
Configuration procedure	165
Verifying the configuration	166
Configuring IPv6 DNS	167
Overview	167
Configuring the IPv6 DNS client	167
Configuring static domain name resolution	167
Configuring dynamic domain name resolution	167
Setting the DSCP value for IPv6 DNS packets	168
Displaying and maintaining IPv6 DNS	168
Static domain name resolution configuration example	169
Network requirements	169
Configuration procedure	169
Dynamic domain name resolution configuration example	170
Network requirements	170
Configuration procedure	170
Verifying the configuration	173
Index	175

Configuring ARP

Overview

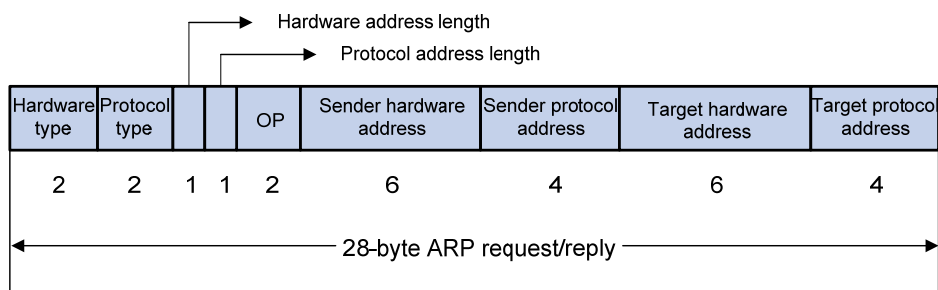
The Address Resolution Protocol (ARP) is used to resolve an IP address into a physical address (Ethernet MAC address, for example).

In an Ethernet LAN, a device uses ARP to resolve the IP address of the next hop to the corresponding MAC address.

ARP message format

ARP messages include ARP requests and ARP replies. Figure 1 shows the format of the ARP request/reply. Numbers in the figure refer to field lengths.

Figure 1 ARP message format



ARP message fields:

- **Hardware type**—The hardware address type. Value 1 represents Ethernet.
- **Protocol type**—The type of the protocol address to be mapped. The hexadecimal value 0x0800 represents IP.
- **Hardware address length and protocol address length**—Length, in bytes, of a hardware address and a protocol address. For an Ethernet address, the value of the hardware address length field is 6. For an IPv4 address, the value of the protocol address length field is 4.
- **OP**—Operation code, which describes type of the ARP message. Value 1 represents an ARP request, and value 2 represents an ARP reply.
- **Sender hardware address**—Hardware address of the device sending the message.
- **Sender protocol address**—Protocol address of the device sending the message.
- **Target hardware address**—Hardware address of the device to which the message is being sent.
- **Target protocol address**—Protocol address of the device to which the message is being sent.

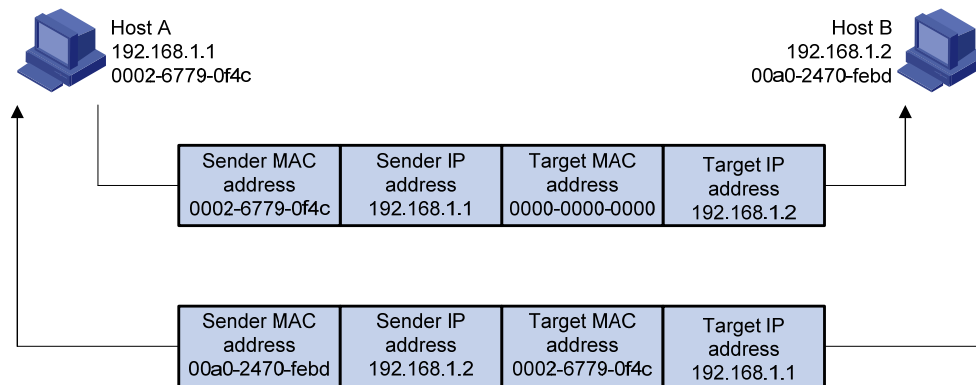
ARP operation

If Host A and Host B are on the same subnet and Host A sends a packet to Host B, as shown in Figure 2, the resolution process is:

1. Host A looks in its ARP table to see whether there is an ARP entry for Host B. If yes, Host A uses the MAC address in the entry to encapsulate the IP packet into a data link layer frame and sends the frame to Host B.
2. If Host A finds no entry for Host B, Host A buffers the packet and broadcasts an ARP request using the following information:
 - **Source IP address and source MAC address**—Host A's own IP address and the MAC address
 - **Target IP address**—Host B's IP address
 - **Target MAC address**—An all-zero MAC address

All hosts on this subnet can receive the broadcast request, but only the requested host (Host B) processes the request.
3. Host B compares its own IP address with the target IP address in the ARP request. If they are the same, Host B:
 - a. Adds the sender IP address and sender MAC address into its ARP table.
 - b. Encapsulates its MAC address into an ARP reply.
 - c. Unicasts the ARP reply to Host A.
4. After receiving the ARP reply, Host A:
 - a. Adds the MAC address of Host B to its ARP table.
 - b. Encapsulates the MAC address into the packet and sends it to Host B.

Figure 2 ARP address resolution process



If Host A and Host B are on different subnets, the resolution process is as follows:

1. Host A sends an ARP request to the gateway. The target IP address in the ARP request is the IP address of the gateway.
2. After obtaining the MAC address of the gateway from an ARP reply, Host A sends the packet to the gateway.
3. If the gateway maintains the ARP entry of Host B, it forwards the packet to Host B directly; if not, it broadcasts an ARP request, in which the target IP address is the IP address of Host B.
4. After obtaining the MAC address of Host B, the gateway sends the packet to Host B.

ARP table

An ARP table stores dynamic and static ARP entries.

Dynamic ARP entry

ARP automatically creates and updates dynamic entries. A dynamic ARP entry is removed when its aging timer expires or the output interface goes down, and it can be overwritten by a static ARP entry.

Static ARP entry

A static ARP entry is manually configured and maintained. It does not age out, and cannot be overwritten by a dynamic ARP entry.

Static ARP entries protect communication between devices, because attack packets cannot modify the IP-to-MAC mapping in a static ARP entry.

Static ARP entries can be classified into long and short ARP entries.

- To configure a long static ARP entry, specify the IP address, MAC address, VLAN, and output interface. A long static ARP entry is directly used for forwarding matching packets. To allow communication with a host using a fixed IP-to-MAC mapping through a specific interface in a specific VLAN, configure a long static ARP entry for it.
- To configure a short static ARP entry, you only need to specify the IP address and MAC address. The device first sends an ARP request whose target IP address is the IP address of the short entry. If the sender IP and MAC addresses in the received ARP reply match the IP and MAC addresses of the short static ARP entry, the device adds the interface receiving the ARP reply to the short static ARP entry, and then the entry can be used for forwarding the matching IP packets.

To communicate with a host by using a fixed IP-to-MAC mapping, configure a short static ARP entry for it.

Configuring a static ARP entry

A static ARP entry is effective when the device it corresponds to works properly. However, when a VLAN or VLAN interface is deleted, any static ARP entry corresponding to it will also be deleted (if it is a long static ARP entry) or will become unresolved (if it is a short and resolved static ARP entry).

Follow these guidelines when you configure a long static ARP entry:

- The *vlan-id* argument must be the ID of an existing VLAN where the ARP entry resides. The specified Ethernet interface must belong to that VLAN. The VLAN interface of the VLAN must be created.
- The IP address of the VLAN interface of the VLAN specified by the *vlan-id* argument must belong to the same subnet as the IP address specified by the *ip-address* argument.

To configure a static ARP entry:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Configure a static ARP entry.	<ul style="list-style-type: none">• Configure a long static ARP entry: arp static ip-address mac-address vlan-id interface-type interface-number• Configure a short static ARP entry: arp static ip-address mac-address	Use either command.

Configuring the maximum number of dynamic ARP entries for an interface

An interface can dynamically learn ARP entries. To prevent an interface from holding too many ARP entries, you can set the maximum number of dynamic ARP entries that an interface can learn. When the maximum number is reached, the interface stops learning ARP entries.

A Layer 2 interface can learn an ARP entry only when both its maximum number and the VLAN interface's maximum number are not reached.

To set the maximum number of dynamic ARP entries that an interface can learn:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter Ethernet interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Set the maximum number of dynamic ARP entries that the interface can learn.	arp max-learning-num <i>number</i>	Optional. By default, a Layer 2 interface does not limit the number of dynamic ARP entries. A Layer 3 interface can learn up to 1024 dynamic ARP entries. If the value of the <i>number</i> argument is set to 0, the interface is disabled from learning dynamic ARP entries.

Setting the aging timer for dynamic ARP entries

Each dynamic ARP entry in the ARP table has a limited lifetime, called aging timer. The aging timer of a dynamic ARP entry is reset each time the dynamic ARP entry is updated. Dynamic ARP entries that are not updated before their aging timers expire are deleted from the ARP table.

To set the age timer for dynamic ARP entries:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Set the age timer for dynamic ARP entries.	arp timer aging <i>aging-time</i>	Optional. 20 minutes by default.

Enabling dynamic ARP entry check

The dynamic ARP entry check function controls whether the device supports dynamic ARP entries with multicast MAC addresses.

When dynamic ARP entry check is enabled, the device cannot learn dynamic ARP entries containing multicast MAC addresses.

When dynamic ARP entry check is disabled, the device can learn dynamic ARP entries containing multicast MAC addresses.

To enable dynamic ARP entry check:

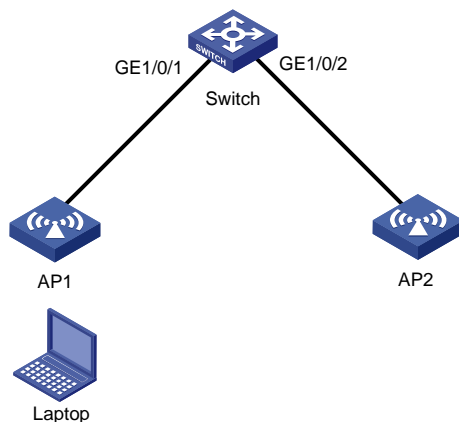
Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable dynamic ARP entry check.	arp check enable	Optional. Enabled by default.

Configuring ARP quick update

H3C recommends you enable ARP quick update in WLAN networks only.

As shown in [Figure 3](#), the laptop frequently roams between AP 1 and AP 2. This affects the mapping between its MAC address and output interface on the switch. If the switch does not update its ARP table immediately after the output interface changes, it might fail to communicate with the laptop.

Figure 3 ARP quick update application scenario



With ARP quick update enabled, the switch updates the corresponding ARP entry immediately after the change of the mapping between a MAC address and an output interface to ensure nonstop data forwarding.

To enable ARP quick update:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable ARP quick update.	mac-address station-move quick-notify enable	Optional. Disabled by default.

Configuring multicast ARP

Microsoft Network Load Balancing (NLB) is a load balancing technology for server clustering developed on Windows Server.

NLB supports load sharing and redundancy among servers within a cluster. To implement fast failover, NLB requires that the switch forwards network traffic to all servers or specified servers in the cluster, and each server filters out unexpected traffic. In a medium or small data center that uses the Windows Server operating system, the proper cooperation of the switch and NLB is very important. For more information about NLB, see the related documents of Windows Sever.

Microsoft NLB provides the following packet sending modes to make the switch forward network traffic to all servers or specified servers:

- **Unicast mode**—NLB assigns each cluster member a common MAC address, which is the cluster MAC address, and changes the source MAC address of each sent packet. Thus, the switch cannot add the cluster MAC address to its MAC table. In addition, because the cluster MAC address is unknown to the switch, packets destined to it are forwarded on all the ports of the switch.
- **Multicast mode**—NLB uses a multicast MAC address that is a virtual MAC address for network communication, for example 0300-5e11-1111.
- **Internet Group Management Protocol (IGMP) multicast mode**—The switch sends packets only out of the ports that connect to the cluster members rather than all ports.

NOTE:

Multicast ARP is applicable to only multicast-mode NLB.

To configure multicast ARP:

Step	Command	Remarks
1. Disable the ARP entry check function.	undo arp check enable	N/A
2. Configure a static ARP entry.	arp static ip-address mac-address vlan-id interface-type interface-number	Optional.
3. Configure a static multicast MAC address entry.	mac-address multicast mac-address interface interface-list vlan vlan-id	See <i>IP Multicast Command Reference</i> .

Displaying and maintaining ARP

CAUTION:

Clearing ARP entries from the ARP table might cause communication failures.

Task	Command	Remarks
Display ARP entries in the ARP table .	display arp [[all dynamic static] [slot slot-number] vlan vlan-id interface interface-type interface-number] [count] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display the ARP entry for a specified IP address.	display arp ip-address [slot slot-number] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display the age timer for dynamic ARP entries.	display arp timer aging [{ begin exclude include } <i>regular-expression</i>]	Available in any view

Task	Command	Remarks
Clear ARP entries from the ARP table .	reset arp { all dynamic static slot slot-number interface interface-type interface-number }	Available in user view

ARP configuration examples

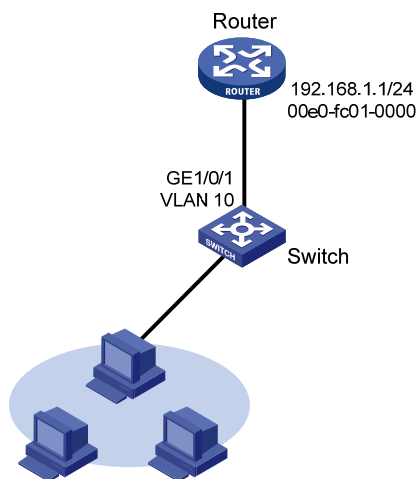
Static ARP entry configuration example

Network requirements

As shown in Figure 4, hosts are connected to the switch, which is connected to the router through interface GigabitEthernet 1/0/1 in VLAN 10. The IP and MAC addresses of the router are 192.168.1.1/24 and 00e0-fc01-0000 respectively.

To prevent malicious users from attacking the switch and enhance security for communications between the router and switch, configure a static ARP entry for the router on the switch.

Figure 4 Network diagram



Configuration procedure

Configure the switch:

Create VLAN 10.

```
<Switch> system-view
[Switch] vlan 10
[Switch-vlan10] quit
```

Add interface GigabitEthernet 1/0/1 to VLAN 10.

```
[Switch] interface GigabitEthernet 1/0/1
[Switch-GigabitEthernet1/0/1] port link-type trunk
[Switch-GigabitEthernet1/0/1] port trunk permit vlan 10
[Switch-GigabitEthernet1/0/1] quit
```

Create interface VLAN-interface 10 and configure its IP address.

```
[Switch] interface vlan-interface 10
```

```
[Switch-vlan-interface10] ip address 192.168.1.2 24
```

```
[Switch-vlan-interface10] quit
```

Configure a static ARP entry that has IP address 192.168.1.1, MAC address 00e0-fc01-0000, and output interface GigabitEthernet 1/0/1 in VLAN 10.

```
[Switch] arp static 192.168.1.1 00e0-fc01-0000 10 GigabitEthernet 1/0/1
```

Display information about static ARP entries.

```
[Switch] display arp static
```

IP Address	MAC Address	VLAN ID	Interface	Aging Type
192.168.1.1	00e0-fc01-0000	10	GE1/0/1	N/A S

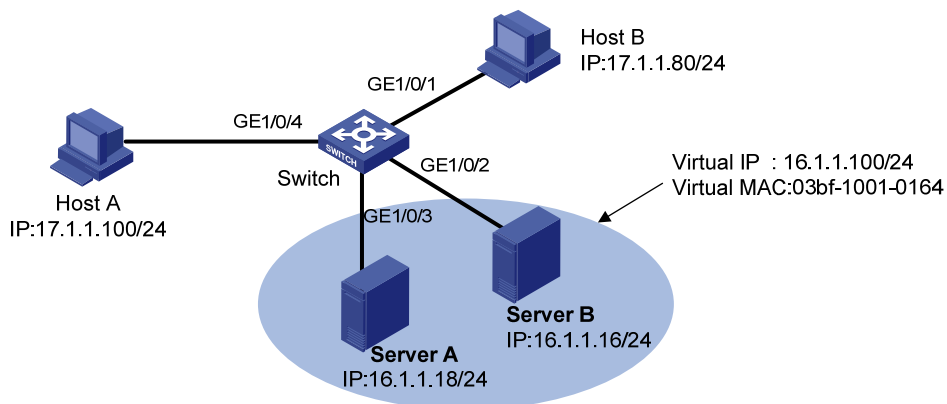
Multicast ARP configuration example

Network requirements

As shown in [Figure 5](#), a small data center uses Microsoft multicast-mode NLB. To enable the switches to cooperate with NLB, configure the following:

- Add GigabitEthernet 1/0/2 and GigabitEthernet 1/0/3 into VLAN 1, and specify IP address 16.1.1.30/24 for VLAN-interface 1.
- Add GigabitEthernet 1/0/1 and GigabitEthernet 1/0/4 into VLAN 2, and specify IP address 17.1.1.1/24 for VLAN-interface 2.
- Specify 17.1.1.1/24 as the default gateway of Host A and Host B.
- Specify 16.1.1.30/24 as the default gateway of Server A and Server B.
- Disable the ARP entry check function so that the switch can learn dynamic ARP entries containing multicast MAC addresses.
- Configure a static multicast MAC address entry so that only interfaces GigabitEthernet 1/0/2 and GigabitEthernet 1/0/3 can receive multicast information.

Figure 5 Network diagram



Configuration procedure

This example only describes multicast ARP configuration on the switch, and is only applicable to multicast NLB. For NLB configuration on the servers, see the related documents of the Windows Server.

Specify an IP address for VLAN-interface 2.

```
<Switch> system-view
```

```
[Switch] vlan 2
```

```

[Switch-vlan2] port GigabitEthernet 1/0/4
[Switch-vlan2] port GigabitEthernet 1/0/1
[Switch-vlan2] quit
[Switch] interface vlan-interface 2
[Switch-Vlan-interface2] ip address 17.1.1.1 255.255.255.0
[Switch-Vlan-interface2] quit

# Specify an IP address for VLAN-interface 1.
[Switch] interface vlan-interface 1
[Switch-Vlan-interface1] ip address 16.1.1.30 255.255.255.0
[Switch-Vlan-interface1] quit

# Disable the ARP entry check function.
[Switch] undo arp check enable

# Configure a static multicast MAC address entry.
[Switch] mac-address multicast 03bf-1001-0164 interface GigabitEthernet 1/0/2 Gigabi
tEthernet 1/0/3 vlan 1

```

Verifying the configuration

- **NLB load sharing**—Enables the FTP server function of Server A and Server B. Host A and Host B send requests to the virtual IP address and each of them logs in to a different server.
- **NLB redundancy**—Disables the network interface card of Server A. Host A and Host B send requests to the virtual IP address and both log in to the FTP server on Server B.

Configuring gratuitous ARP

Overview

In a gratuitous ARP packet, the sender IP address and the target IP address are the IP address of the sending device.

A device sends a gratuitous ARP packet for either of the following purposes:

- Determine whether its IP address is already used by another device. If the IP address is already used, the device is informed of the conflict by an ARP reply.
- Inform other devices of a change of its MAC address.

Gratuitous ARP packet learning

This feature enables a device to create or update ARP entries by using the sender IP and MAC addresses in received gratuitous ARP packets.

With this feature disabled, the device uses received gratuitous ARP packets to update existing ARP entries only.

Periodic sending of gratuitous ARP packets

Enabling a device to periodically send gratuitous ARP packets helps downstream devices update their corresponding ARP entries or MAC entries in time. This feature can be used to:

- Prevent gateway spoofing.

When an attacker sends forged gratuitous ARP packets to the hosts on a network, the traffic destined for the gateway from the hosts is sent to the attacker instead. As a result, the hosts cannot access the external network.

To prevent gateway spoofing attacks, enable the gateway to send gratuitous ARP packets containing its primary IP address and manually configured secondary IP addresses at a specific interval, so hosts can learn correct gateway address information.
- Prevent ARP entries from aging out.

If network traffic is heavy or if a host's CPU usage is high on a host, received ARP packets might be discarded or not be processed in time. Eventually, the dynamic ARP entries on the receiving host age out, and the traffic between the host and the corresponding devices is interrupted until the host re-creates the ARP entries.

To prevent this problem, enable the gateway to send gratuitous ARP packets periodically. The gratuitous ARP packets contain the gateway's primary IP address or one of its manually configured secondary IP addresses, so the receiving host can update ARP entries in time, ensuring traffic continuity.

Configuration guidelines

Follow these guidelines when you configure gratuitous ARP:

- You can enable periodic sending of gratuitous ARP packets in VLAN interface view.

- You can enable periodic sending of gratuitous ARP packets on a maximum of 1024 interfaces.
- Periodic sending of gratuitous ARP packets takes effect only when the link of the enabled interface goes up and an IP address has been assigned to the interface.
- If you change the interval for sending gratuitous ARP packets, the configuration is effective at the next sending interval.
- The frequency of sending gratuitous ARP packets might be much lower than is expected if this function is enabled on multiple interfaces, if each interface is configured with multiple secondary IP addresses, or if a small sending interval is configured in such cases.

Configuration procedure

To configure gratuitous ARP:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable learning of gratuitous ARP packets.	gratuitous-arp-learning enable	Optional. Enabled by default.
3. Enable the device to send gratuitous ARP packets upon receiving ARP requests from another subnet.	gratuitous-arp-sending enable	By default, a device does not send gratuitous ARP packets upon receiving ARP requests from another subnet.
4. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
5. Enable periodic sending of gratuitous ARP packets and set the sending interval.	arp send-gratuitous-arp [interval <i>milliseconds</i>]	Disabled by default.

Enabling IP conflict notification

If the sender IP address of a received gratuitous ARP packet is being used by the receiving device, by default, the receiving device sends a gratuitous ARP request, and it displays an error message after it receives an ARP reply. The receiving device repeats the default processing 5 seconds after displaying the error message, and it stops the processing when the conflict is resolved.

You can use this command to enable the device to display error message without sending any gratuitous ARP request for conflict confirmation. The receiving device displays the message every 30 seconds until the conflict is resolved.

To enable IP conflict notification:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable IP conflict notification.	arp ip-conflict prompt	Optional. Disabled by default.

Configuring proxy ARP

Overview

Proxy ARP enables a device on a network to answer ARP requests for an IP address not on that network. With proxy ARP, hosts on different broadcast domains can communicate with each other as they do on the same network.

Proxy ARP includes common proxy ARP and local proxy ARP.

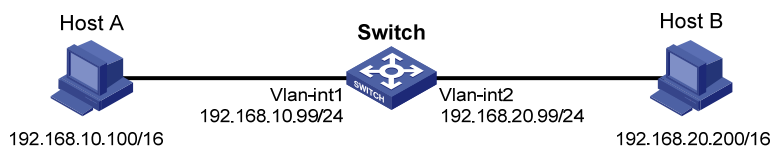
- **Common proxy ARP**—Allows communication between hosts that connect to different Layer-3 interfaces and reside in different broadcast domains.
- **Local proxy ARP**—Allows communication between hosts that connect to the same Layer-3 interface and reside in different broadcast domains.

Common proxy ARP

A common proxy ARP enabled device allows hosts that reside on different subnets to communicate.

As shown in [Figure 6](#), Switch connects to two subnets through VLAN-interface 1 and VLAN-interface 2. The IP addresses of the two interfaces are 192.168.10.99/24 and 192.168.20.99/24. Host A and Host B are assigned the same prefix 192.168.0.0. Host A connects to VLAN-interface 1 and Host B connects to VLAN-interface 2.

Figure 6 Application environment of common proxy ARP



Because Host A and Host B have the same prefix 192.168.0.0, Host A considers that Host B is on the same network, and it broadcasts an ARP request for the MAC address of Host B. However, Host B cannot receive this request because it is in a different broadcast domain.

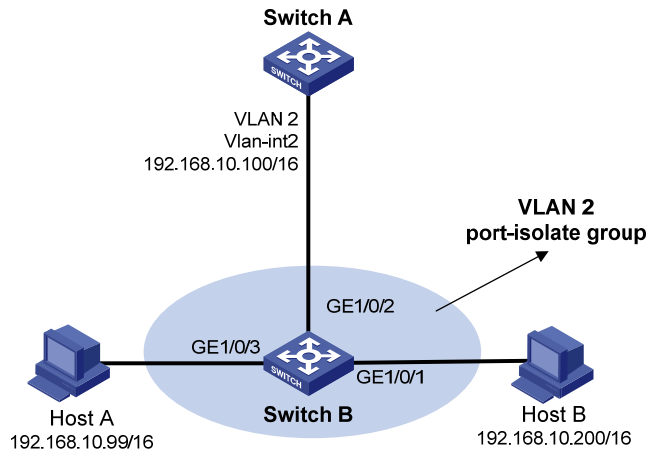
You can common enable proxy ARP on VLAN-interface 1 of the switch so that the switch can reply to the ARP request from Host A with the MAC address of VLAN-interface 1, and forward packets sent from Host A to Host B. In this case, the switch acts as a proxy of Host B.

A main advantage of common proxy ARP is that you can enable it on a single switch without disturbing routing tables of other routers in the network. Proxy ARP acts as the gateway for hosts that are not configured with a default gateway or do not have routing capability.

Local proxy ARP

As shown in [Figure 7](#), Host A and Host B belong to VLAN 2, but are isolated at Layer 2. Host A connects to GigabitEthernet 1/0/3 while Host B connects to GigabitEthernet 1/0/1. Enable local proxy ARP on Switch A to allow Layer 3 communication between the two hosts.

Figure 7 Application environment of local proxy ARP



Enable local proxy ARP in one of the following cases:

- Hosts connecting to different isolated Layer 2 ports in the same VLAN need to communicate at Layer 3.
- If an isolate-user-VLAN is configured, hosts in different secondary VLANs of the isolate-user-VLAN need to communicate at Layer 3.

Enabling common proxy ARP

To enable common proxy ARP in VLAN interface view:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type interface-number</i>	N/A
3. Enable proxy ARP.	proxy-arp enable	Disabled by default

Enabling local proxy ARP

To enable local proxy ARP in VLAN interface view:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type interface-number</i>	N/A
3. Enable local proxy ARP.	local-proxy-arp enable [ip-range <i>startIP to endIP</i>]	Disabled by default

Displaying and maintaining proxy ARP

Task	Command	Remarks
Display whether common proxy ARP is enabled.	display proxy-arp [interface <i>interface-type interface-number</i>] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display whether local proxy ARP is enabled.	display local-proxy-arp [interface <i>interface-type interface-number</i>] [{ begin exclude include } <i>regular-expression</i>]	Available in any view

Proxy ARP configuration examples

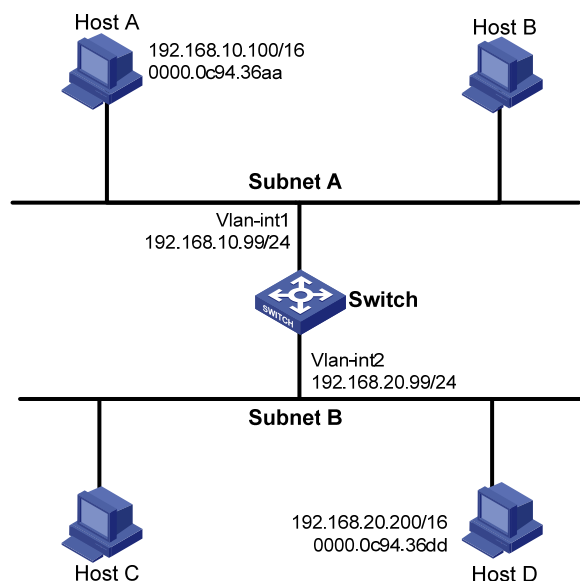
Common proxy ARP configuration example

Network requirements

As shown in Figure 8, Host A and Host D have the same IP prefix and mask (IP addresses of Host A and Host D are 192.168.10.100/16 and 192.168.20.200/16 respectively), but they are located on different subnets separated by the switch (Host A belongs to VLAN 1 while Host D belongs to VLAN 2). As a result, Host D cannot receive or respond to any ARP request from Host A.

You must configure proxy ARP on the switch to enable communication between the two hosts.

Figure 8 Network diagram



Configuration procedure

```
# Create VLAN 2.
<Switch> system-view
[Switch] vlan 2
[Switch-vlan2] quit

# Specify the IP address of interface VLAN-interface 1.
[Switch] interface vlan-interface 1
[Switch-Vlan-interface1] ip address 192.168.10.99 255.255.255.0
```

```

# Enable proxy ARP on interface VLAN-interface 1.
[Switch-Vlan-interface1] proxy-arp enable
[Switch-Vlan-interface1] quit

# Specify the IP address of interface VLAN-interface 2.
[Switch] interface vlan-interface 2
[Switch-Vlan-interface2] ip address 192.168.20.99 255.255.255.0

# Enable proxy ARP on interface VLAN-interface 2.
[Switch-Vlan-interface2] proxy-arp enable

```

After completing preceding configurations, use the **ping** command to verify the connectivity between Host A and Host D.

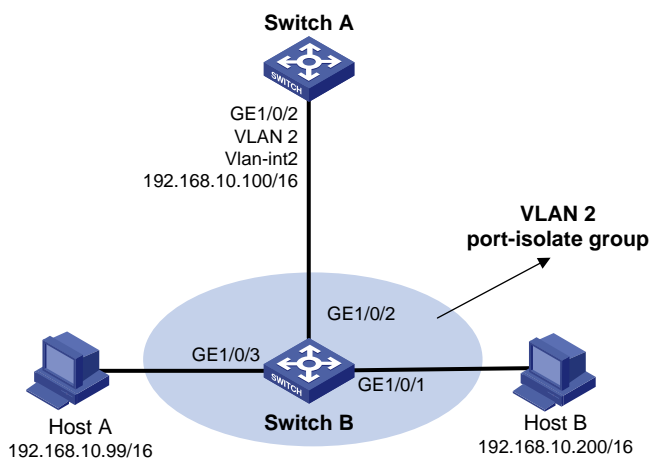
Local proxy ARP configuration example in case of port isolation

Network requirements

As shown in Figure 9, Host A and Host B belong to the same VLAN, and connect to Switch B via GigabitEthernet 1/0/3 and GigabitEthernet 1/0/1 respectively. Switch B connects to Switch A via GigabitEthernet 1/0/2.

Configure port isolation on GigabitEthernet 1/0/3 and GigabitEthernet 1/0/1 of Switch B to isolate Host A from Host B at Layer 2. Enable local proxy ARP on Switch A to allow communication between Host A and Host B at Layer 3.

Figure 9 Network diagram



Configuration procedure

1. Configure Switch B:

```

# Add GigabitEthernet 1/0/3, GigabitEthernet 1/0/1 and GigabitEthernet 1/0/2 to VLAN 2.
Configure port isolation on Host A and Host B.

```

```

<SwitchB> system-view
[SwitchB] vlan 2
[SwitchB-vlan2] port GigabitEthernet 1/0/3
[SwitchB-vlan2] port GigabitEthernet 1/0/1
[SwitchB-vlan2] port GigabitEthernet 1/0/2
[SwitchB-vlan2] quit
[SwitchB] interface GigabitEthernet 1/0/3

```

```
[SwitchB-GigabitEthernet1/0/3] port-isolate enable
[SwitchB-GigabitEthernet1/0/3] quit
[SwitchB] interface GigabitEthernet 1/0/1
[SwitchB-GigabitEthernet1/0/1] port-isolate enable
[SwitchB-GigabitEthernet1/0/1] quit
```

2. Configure Switch A:

Create VLAN 2, and add GigabitEthernet 1/0/2 to VLAN 2.

```
<SwitchA> system-view
[SwitchA] vlan 2
[SwitchA-vlan2] port GigabitEthernet 1/0/2
[SwitchA-vlan2] quit
[SwitchA] interface vlan-interface 2
[SwitchA-Vlan-interface2] ip address 192.168.10.100 255.255.0.0
```

From Host A, ping Host B. The ping operation is unsuccessful because they are isolated at Layer 2.

Configure local proxy ARP to allow communication between Host A and Host B at Layer 3.

```
[SwitchA-Vlan-interface2] local-proxy-arp enable
```

From Host A, ping Host B. The ping operation is successful after the configuration.

Local proxy ARP configuration example in isolate-user-VLAN

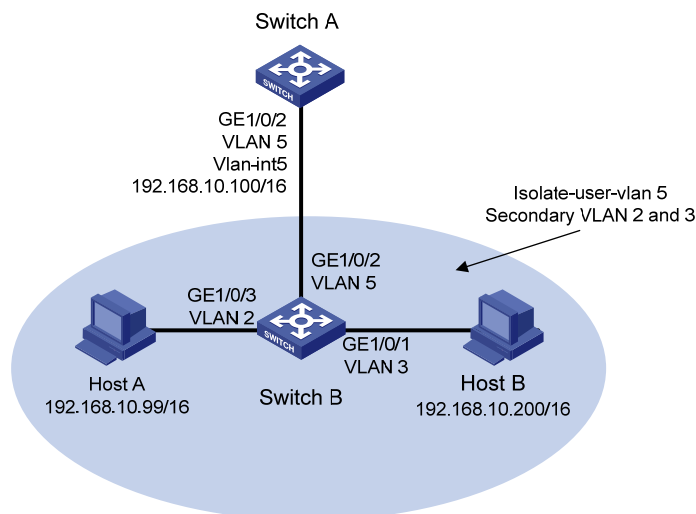
Network requirements

As shown in [Figure 10](#), Switch B is attached to Switch A. VLAN 5 on Switch B is an isolate-user-VLAN, which includes uplink port GigabitEthernet 1/0/2 and two secondary VLANs, VLAN 2 and VLAN 3. GigabitEthernet 1/0/3 belongs to VLAN 2, and GigabitEthernet 1/0/1 belongs to VLAN 3.

Host A belongs to VLAN 2 and connects to GigabitEthernet 1/0/3 of Switch B. Host B belongs to VLAN 3 and connects to GigabitEthernet 1/0/1 of Switch B.

As Host A and Host B belong to different secondary VLANs, they are isolated at Layer 2. Configure local proxy ARP on Switch A to implement Layer 3 communication between Host A and Host B.

Figure 10 Network diagram



Configuration procedure

1. Configure Switch B:

Create VLAN 2, VLAN 3, and VLAN 5 on Switch B. Add GigabitEthernet 1/0/3 to VLAN 2, GigabitEthernet 1/0/1 to VLAN 3, and GigabitEthernet 1/0/2 to VLAN 5. Configure VLAN 5 as the isolate-user-VLAN, and VLAN 2 and VLAN 3 as secondary VLANs. Configure the mappings between isolate-user-VLAN and the secondary VLANs.

```
<SwitchB> system-view
[SwitchB] vlan 2
[SwitchB-vlan2] port GigabitEthernet 1/0/3
[SwitchB-vlan2] quit
[SwitchB] vlan 3
[SwitchB-vlan3] port GigabitEthernet 1/0/1
[SwitchB-vlan3] quit
[SwitchB] vlan 5
[SwitchB-vlan5] port GigabitEthernet 1/0/2
[SwitchB-vlan5] isolate-user-vlan enable
[SwitchB-vlan5] quit
[SwitchB] interface GigabitEthernet 1/0/2
[SwitchB-GigabitEthernet1/0/2] port isolate-user-vlan 5 promiscuous
[SwitchB-GigabitEthernet1/0/2] quit
[SwitchB] interface GigabitEthernet 1/0/1
[SwitchB-GigabitEthernet1/0/1] port isolate-user-vlan host
[SwitchB-GigabitEthernet1/0/1] quit
[SwitchB] interface GigabitEthernet 1/0/3
[SwitchB-GigabitEthernet1/0/3] port isolate-user-vlan host
[SwitchB-GigabitEthernet1/0/3] quit
[SwitchB] isolate-user-vlan 5 secondary 2 3
```

2. Configure Switch A:

Create VLAN 5 and add GigabitEthernet 1/0/2 to it.

```
<SwitchA> system-view
[SwitchA] vlan 5
[SwitchA-vlan5] port GigabitEthernet 1/0/2
[SwitchA-vlan5] quit
[SwitchA] interface vlan-interface 5
[SwitchA-Vlan-interface5] ip address 192.168.10.100 255.255.0.0
```

From Host A, ping Host B. The ping operation is unsuccessful because they are isolated at Layer 2.

Configure local proxy ARP to implement Layer 3 communication between Host A and Host B.

```
[SwitchA-Vlan-interface5] local-proxy-arp enable
```

From Host A, ping Host B. The ping operation is successful after the configuration.

Configuring ARP snooping

Overview

The ARP snooping feature is used in Layer 2 switching networks. It creates ARP snooping entries using ARP packets.

If ARP snooping is enabled on a VLAN of a device, ARP packets received by the interfaces of the VLAN are redirected to the CPU. The CPU uses ARP packets to create ARP snooping entries comprising source IP and MAC addresses, VLAN and receiving port information.

The aging time and valid period of an ARP snooping entry are 25 minutes and 15 minutes, respectively. If an ARP snooping entry is not updated within 15 minutes, it becomes invalid and cannot be used. After that, if an ARP packet whose source IP and MAC addresses correspond with the entry is received, the entry becomes valid, and its age timer restarts. If the age timer of an ARP entry expires, the entry is removed.

If the ARP snooping device receives an ARP packet that has the same sender IP address as but a different sender MAC address from a valid ARP snooping entry, it considers that an attack occurs. An ARP snooping entry conflict occurs in this case. As a result, the ARP snooping entry becomes invalid and is removed after 25 minutes.

Configuration procedure

To enable ARP snooping for a VLAN:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter VLAN view.	vlan <i>vlan-id</i>	N/A
3. Enable ARP snooping.	arp-snooping enable	Disabled by default

Displaying and maintaining ARP snooping

Task	Command	Remarks
Display ARP snooping entries.	display arp-snooping [ip <i>ip-address</i> vlan <i>vlan-id</i>] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Remove ARP snooping entries.	reset arp-snooping [ip <i>ip-address</i> vlan <i>vlan-id</i>]	Available in user view

Configuring IP addressing

This chapter describes IP addressing basic and manual IP address assignment for interfaces. Dynamic IP address assignment (BOOTP and DHCP) are beyond the scope of this chapter.

Overview

This section describes the IP addressing basics.

IP addressing uses a 32-bit address to identify each host on a network. To make addresses easier to read, they are written in dotted decimal notation, each address being four octets in length. For example, address 00001010000000010000000100000001 in binary is written as 10.1.1.1.

IP address classes

Each IP address breaks down into two parts:

- **Net ID**—Identifies a network. The first several bits of a net ID, known as the class field or class bits, identify the class of the IP address.
- **Host ID**—Identifies a host on a network.

IP addresses are divided into five classes, shown in [Figure 11](#). The shaded areas represent the address class. The first three classes are widely used.

Figure 11 IP address classes

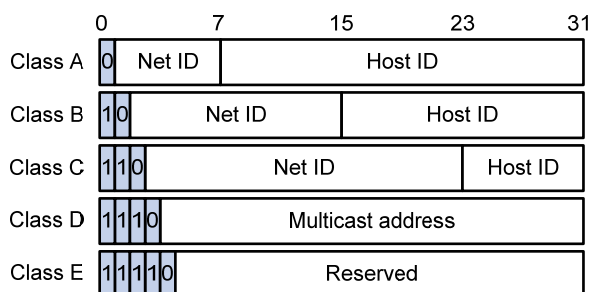


Table 1 IP address classes and ranges

Class	Address range	Remarks
A	0.0.0.0 to 127.255.255.255	The IP address 0.0.0.0 is used by a host at startup for temporary communication. This address is never a valid destination address. Addresses starting with 127 are reserved for loopback test. Packets destined to these addresses are processed locally as input packets rather than sent to the link.
B	128.0.0.0 to 191.255.255.255	N/A
C	192.0.0.0 to 223.255.255.255	N/A

Class	Address range	Remarks
D	224.0.0.0 to 239.255.255.255	Multicast addresses.
E	240.0.0.0 to 255.255.255.255	Reserved for future use except for the broadcast address 255.255.255.255.

Special IP addresses

The following IP addresses are for special use and cannot be used as host IP addresses.

- **IP address with an all-zero net ID**—Identifies a host on the local network. For example, IP address 0.0.0.16 indicates the host with a host ID of 16 on the local network.
- **IP address with an all-zero host ID**—Identifies a network.
- **IP address with an all-one host ID**—Identifies a directed broadcast address. For example, a packet with the destination address of 192.168.1.255 will be broadcast to all the hosts on the network 192.168.1.0.

Subnetting and masking

Subnetting divides a network down into smaller networks called subnets by using some bits of the host ID to create a subnet ID.

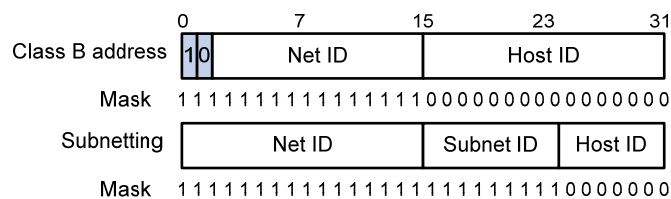
Masking identifies the boundary between the host ID and the combination of net ID and subnet ID. (When subnetting is not adopted, a mask identifies the boundary between the net ID and the host ID.)

Each subnet mask is made up of 32 bits that correspond to the bits in an IP address. In a subnet mask, consecutive ones represent the net ID and subnet ID, and consecutive zeros represent the host ID.

Before being subnetted, Class A, B, and C networks use the following default masks (also called natural masks): 255.0.0.0, 255.255.0.0, and 255.255.255.0 respectively.

Figure 12 shows how a Class B network is subnetted.

Figure 12 Subnetting a Class B network



Subnetting increases the number of addresses that cannot be assigned to hosts. After being subnetted, a network can accommodate fewer hosts.

For example, a Class B network without subnetting can accommodate 1022 more hosts than the same network subnetted into 512 subnets.

- **Without subnetting**—65,534 hosts ($2^{16} - 2$). (The two deducted addresses are the broadcast address, which has an all-one host ID, and the network address, which has an all-zero host ID.)
- **With subnetting**—Using the first 9 bits of the host-id for subnetting provides 512 (2^9) subnets. However, only 7 bits remain available for the host ID. This allows 126 ($2^7 - 2$) hosts in each subnet, a total of 64,512 hosts (512×126).

Assigning an IP address to an interface

You can assign an interface one primary address and multiple secondary addresses.

Generally, you only need to assign the primary address to an interface. In some cases, you need to assign secondary IP addresses to the interface. For example, if the interface connects to two subnets, to enable the device to communicate with all hosts on the LAN, you need to assign a primary IP address and a secondary IP address to the interface.

Configuration guidelines

Follow these guidelines when you assign an IP address to an interface:

- Each interface has only one primary IP address. A newly configured primary IP address overwrites the previous one.
- You cannot assign secondary IP addresses to an interface that obtains an IP address through BOOTP or DHCP.
- The primary and secondary IP addresses you assign to the interface can be located on the same network segment, but different interfaces on your device must reside on different network segments.
- You can manually assign an IP address to an interface, or configure the interface to obtain an IP address through BOOTP or DHCP. If you change the way an interface obtains an IP address, the new IP address overwrites the previous one.
- The switch supports a 31-bit subnet mask (the mask 255.255.255.254) for saving IP addresses in the point-to-point communication.

Configuration procedure

To assign an IP address to an interface:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Assign an IP address to the interface.	ip address <i>ip-address</i> { <i>mask-length</i> <i>mask</i> } [sub]	By default, no IP address is assigned to any interface.

Configuration example

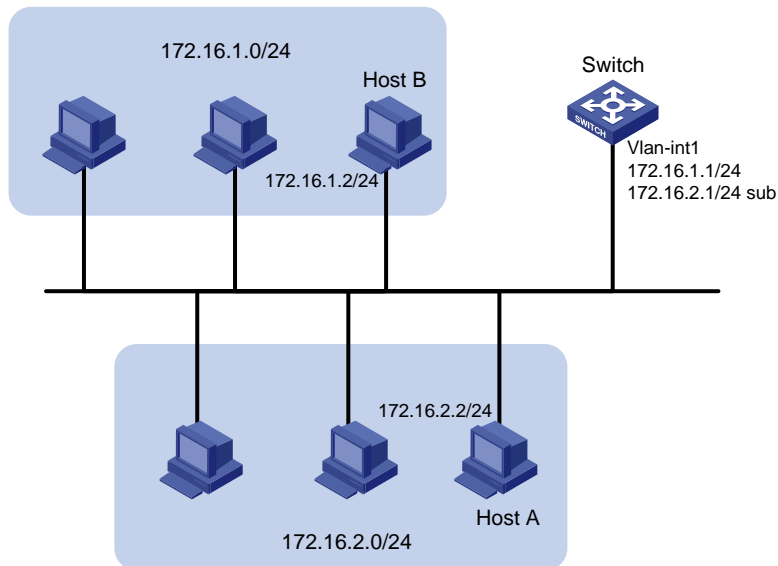
Network requirements

As shown in [Figure 13](#), a port in VLAN 1 on a switch is connected to a LAN comprising two segments: 172.16.1.0/24 and 172.16.2.0/24.

To enable the hosts on the two subnets to communicate with the external network through the switch, and to enable the hosts on the two subnets to communicate with each other:

- Assign a primary IP address and a secondary IP address to VLAN-interface 1 on the switch.
- Set the primary IP address of VLAN-interface 1 as the gateway address of the hosts on subnet 172.16.1.0/24, and the secondary IP address of VLAN-interface 1 as the gateway address of the hosts on subnet 172.16.2.0/24.

Figure 13 Network diagram



Configuration procedure

Assign a primary IP address and a secondary IP address to VLAN-interface 1.

```
<Switch> system-view
[Switch] interface vlan-interface 1
[Switch-Vlan-interface1] ip address 172.16.1.1 255.255.255.0
[Switch-Vlan-interface1] ip address 172.16.2.1 255.255.255.0 sub
```

Set the gateway address to 172.16.1.1 on the hosts attached to subnet 172.16.1.0/24, and to 172.16.2.1 on the hosts attached to subnet 172.16.2.0/24.

From the switch, ping a host on subnet 172.16.1.0/24 to verify the connectivity.

```
<Switch> ping 172.16.1.2
PING 172.16.1.2: 56 data bytes, press CTRL_C to break
  Reply from 172.16.1.2: bytes=56 Sequence=1 ttl=255 time=25 ms
  Reply from 172.16.1.2: bytes=56 Sequence=2 ttl=255 time=27 ms
  Reply from 172.16.1.2: bytes=56 Sequence=3 ttl=255 time=26 ms
  Reply from 172.16.1.2: bytes=56 Sequence=4 ttl=255 time=26 ms
  Reply from 172.16.1.2: bytes=56 Sequence=5 ttl=255 time=26 ms

--- 172.16.1.2 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 25/26/27 ms
```

The output shows that the switch can communicate with the hosts on subnet 172.16.1.0/24.

From the switch, ping a host on subnet 172.16.2.0/24 to verify the connectivity.

```
<Switch> ping 172.16.2.2
PING 172.16.2.2: 56 data bytes, press CTRL_C to break
  Reply from 172.16.2.2: bytes=56 Sequence=1 ttl=255 time=25 ms
  Reply from 172.16.2.2: bytes=56 Sequence=2 ttl=255 time=26 ms
  Reply from 172.16.2.2: bytes=56 Sequence=3 ttl=255 time=26 ms
```

```

Reply from 172.16.2.2: bytes=56 Sequence=4 ttl=255 time=26 ms
Reply from 172.16.2.2: bytes=56 Sequence=5 ttl=255 time=26 ms

--- 172.16.2.2 ping statistics ---
 5 packet(s) transmitted
 5 packet(s) received
 0.00% packet loss
 round-trip min/avg/max = 25/25/26 ms

```

The output shows that the switch can communicate with the hosts on subnet 172.16.2.0/24.

From a host on subnet 172.16.2.0/24, ping a host on subnet 172.16.1.0/24 to verify the connectivity. Host B can be successfully pinged from Host A.

Displaying and maintaining IP addressing

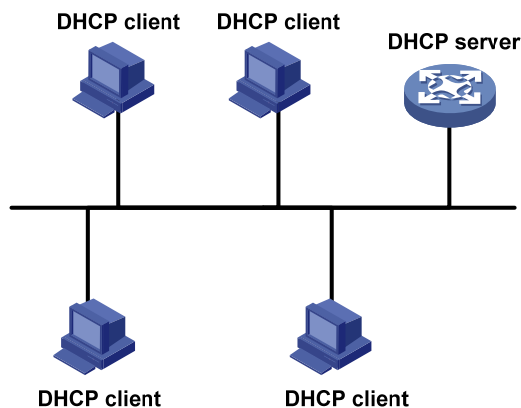
Task	Command	Remarks
Display IP configuration information for a specified Layer 3 interface or all Layer 3 interfaces.	display ip interface [<i>interface-type</i> <i>interface-number</i>] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display brief IP configuration information for a specified Layer 3 interface or all Layer 3 interfaces.	display ip interface [<i>interface-type</i> [<i>interface-number</i>]] brief [{ begin exclude include } <i>regular-expression</i>]	Available in any view

DHCP overview

The Dynamic Host Configuration Protocol (DHCP) provides a framework to assign configuration information to network devices.

DHCP uses the client/server model.

Figure 14 A typical DHCP application



A DHCP client can obtain an IP address and other configuration parameters from a DHCP server on another subnet via a DHCP relay agent. For more information about the DHCP relay agent, see "[Configuring DHCP relay agent.](#)"

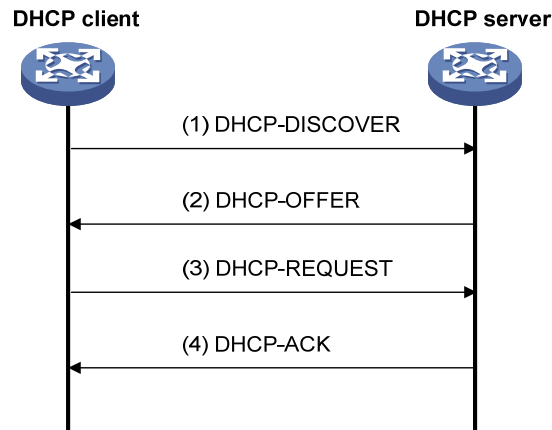
DHCP address allocation

DHCP supports the following mechanisms for IP address allocation.

- **Static allocation**—The network administrator assigns an IP address to a client like a WWW server, and DHCP conveys the assigned address to the client.
- **Automatic allocation**—DHCP assigns a permanent IP address to a client.
- **Dynamic allocation**—DHCP assigns an IP address to a client for a limited period of time, which is called a lease. Most DHCP clients obtain their addresses in this way.

Dynamic IP address allocation process

Figure 15 Dynamic IP address allocation process



1. The client broadcasts a DHCP-DISCOVER message to locate a DHCP server.
2. A DHCP server offers configuration parameters such as an IP address to the client, in a DHCP-OFFER message. The sending mode of the DHCP-OFFER is determined by the flag field in the DHCP-DISCOVER message. For related information, see "[DHCP message format](#)."
3. If several DHCP servers send offers to the client, the client accepts the first received offer, and broadcasts it in a DHCP-REQUEST message to formally request the IP address.
4. All DHCP servers receive the DHCP-REQUEST message, but only the server from which the client accepts the offered IP address returns either a DHCP-ACK message to the client, confirming that the IP address has been allocated to the client, or a DHCP-NAK message, denying the IP address allocation.

After the client receives the DHCP-ACK message, it broadcasts a gratuitous ARP packet to verify whether the IP address assigned by the server is already in use. If the client receives no response within the specified time, the client uses the assigned IP address. Otherwise, the client sends a DHCP-DECLINE message to the server to request an IP address again.

IP addresses offered by other DHCP servers are still assignable to other clients.

IP address lease extension

The IP address dynamically allocated by a DHCP server to a client has a lease. When the lease expires, the IP address is reclaimed by the DHCP server. To continue using the IP address, the client must extend the lease duration.

After half the lease duration, the DHCP client sends a DHCP-REQUEST unicast to the DHCP server to extend the lease. Depending on availability of the IP address, the DHCP server returns either a DHCP-ACK unicast confirming that the client's lease has been extended, or a DHCP-NAK unicast denying the request.

If the client receives no reply, it broadcasts another DHCP-REQUEST message for lease extension after 7/8 lease duration. Again, depending on availability of the IP address, the DHCP server returns either a DHCP-ACK unicast confirming that the client's lease has been extended, or a DHCP-NAK unicast denying the request.

DHCP message format

Figure 16 shows the DHCP message format, which is based on the BOOTP message format although DHCP uses some of the fields in significantly different ways. The numbers in parentheses indicate the size of each field in bytes.

Figure 16 DHCP message format

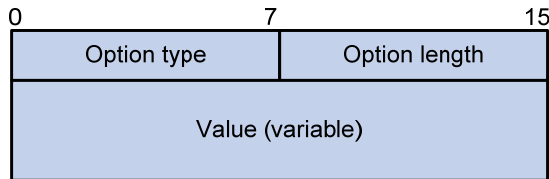
0	7	15	23	31
op (1)		htype (1)		hlen (1)
xid (4)				
secs (2)			flags (2)	
ciaddr (4)				
yiaddr (4)				
siaddr (4)				
giaddr (4)				
chaddr (16)				
sname (64)				
file (128)				
options (variable)				

- **op**—Message type defined in option field. 1 = REQUEST, 2 = REPLY
- **htype, hlen**—Hardware address type and length of a DHCP client.
- **hops**—Number of relay agents a request message traveled.
- **xid**—Transaction ID, a random number chosen by the client to identify an IP address allocation.
- **secs**—Filled in by the client, the number of seconds elapsed since the client began address acquisition or renewal process. Currently this field is reserved and set to 0.
- **flags**—The leftmost bit is defined as the BROADCAST (B) flag. If this flag is set to 0, the DHCP server sent a reply back by unicast. If this flag is set to 1, the DHCP server sent a reply back by broadcast. The remaining bits of the flags field are reserved for future use.
- **ciaddr**—Client IP address if the client has an IP address that is valid and usable. Otherwise, set to zero.
- **yiaddr**—'Your' (client) IP address, assigned by the server.
- **siaddr**—Server IP address, from which the client obtained configuration parameters.
- **giaddr**—(Gateway) IP address of the first relay agent a request message traveled.
- **chaddr**—Client hardware address.
- **sname**—Server host name, from which the client obtained configuration parameters.
- **file**—Bootfile name and path information, defined by the server to the client.
- **options**—Optional parameters field that is variable in length, which includes the message type, lease duration, subnet mask, domain name server IP address, WINS IP address, and other information.

DHCP options

DHCP uses the same message format as BOOTP, but DHCP uses the Option field to carry information for dynamic address allocation and to provide additional configuration information to clients.

Figure 17 DHCP option format



Common DHCP options

The following are common DHCP options:

- **Option 3**—Router option. It specifies the gateway address.
- **Option 6**—DNS server option. It specifies the DNS server's IP address.
- **Option 33**—Static route option. It specifies a list of classful static routes (the destination addresses in these static routes are classful) that a client should add into its routing table. If both Option 33 and Option 121 exist, Option 33 is ignored.
- **Option 51**—IP address lease option.
- **Option 53**—DHCP message type option. It identifies the type of the DHCP message.
- **Option 55**—Parameter request list option. It is used by a DHCP client to request specified configuration parameters. The option contains values that correspond to the parameters requested by the client.
- **Option 60**—Vendor class identifier option. It is used by a DHCP client to identify its vendor, and by a DHCP server to distinguish DHCP clients by vendor class and assign specific IP addresses for the DHCP clients.
- **Option 66**—TFTP server name option. It specifies a TFTP server to be assigned to the client.
- **Option 67**—Bootfile name option. It specifies the bootfile name to be assigned to the client.
- **Option 121**—Classless route option. It specifies a list of classless static routes (the destination addresses in these static routes are classless) that the requesting client should add to its routing table. If both Option 33 and Option 121 exist, Option 33 is ignored.
- **Option 150**—TFTP server IP address option. It specifies the TFTP server IP address to be assigned to the client.

For more information about DHCP options, see RFC 2132 and RFC 3442.

Custom options

Some options, such as Option 43, Option 82, and Option 184, have no unified definitions in RFC 2132.

Vendor-specific option (Option 43)

DHCP servers and clients use Option 43 to exchange vendor-specific configuration information.

The DHCP client can obtain the following information through Option 43:

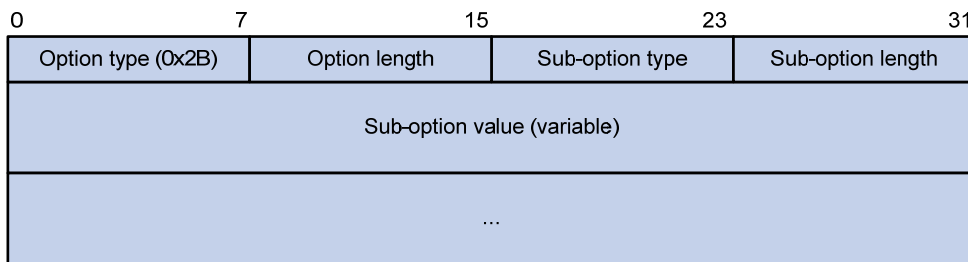
- Auto-Configuration Server (ACS) parameters, including the ACS URL, username, and password.

- Service provider identifier, which is acquired by the Customer Premises Equipment (CPE) from the DHCP server and sent to the ACS for selecting vender-specific configurations and parameters.
- Preboot Execution Environment (PXE) server address, which is used to obtain the bootfile or other control information from the PXE server.

1. Format of Option 43

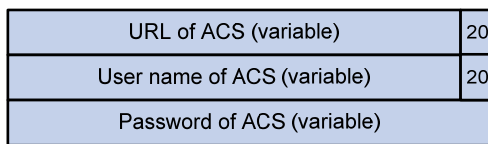
Network configuration parameters are carried in different sub-options of Option 43 as shown in Figure 18.

Figure 18 Option 43 format



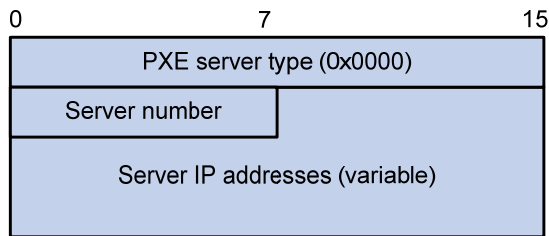
- **Sub-option type**—Type of a sub-option. The field value can be 0x01 (an ACS parameter sub-option), 0x02 (a service provider identifier sub-option), or 0x80 (a PXE server address sub-option).
 - **Sub-option length**—Length of a sub-option excluding the sub-option type and sub-option length fields.
 - **Sub-option value**—Value of a sub-option. The value format varies with sub-options.
2. Format of the sub-option value field of Option 43
- As shown in Figure 19, the value field of the ACS parameter sub-option contains variable ACS URL, ACS username, and ACS password separated by spaces (0x20):

Figure 19 ACS parameter sub-option value field



- The value field of the service provider identifier sub-option contains the service provider identifier.
- Figure 20 shows the format of the value field of the PXE server address sub-option. The value of the PXE server type can only be 0. The server number field indicates the number of PXE servers contained in the sub-option. The server IP addresses field contains the IP addresses of the PXE servers.

Figure 20 PXE server address sub-option value field



Relay agent option (Option 82)

Option 82 is the relay agent option in the option field of the DHCP message. It records the location information about the DHCP client. When a DHCP relay agent or DHCP snooping device receives a client's request, it adds Option 82 to the request message and sends it to the server.

The administrator can locate the DHCP client to further implement security control and accounting. The Option 82 supporting server can also use such information to define individual assignment policies of IP address and other parameters for the clients.

Option 82 can contain up to 255 sub-options and must have one sub-option at least. Option 82 supports two sub-options: sub-option 1 (Circuit ID) and sub-option 2 (Remote ID). DHCP snooping device supports three sub-options: sub-option 1 (Circuit ID), sub-option 2 (Remote ID), and sub-option 9.

Option 82 has no unified definition. Its padding formats vary with vendors.

There are two methods for configuring Option 82:

- **User-defined method**—Manually specify the content of Option 82.
- **Non-user-defined method**—Pad Option 82 in the default normal format, verbose format, private format, or standard format.

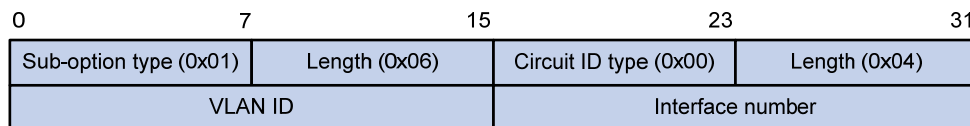
NOTE:

Only the DHCP snooping device supports sub-option 9, padded in either private or standard format.

If you choose normal format and verbose format, you can specify the code type for the sub-options as ASCII or HEX.

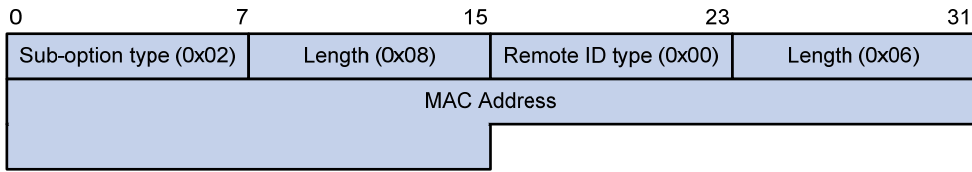
- Normal padding format
 - **Sub-option 1**—Contains the VLAN ID and interface number of the interface that received the client's request. The value of the sub-option type is 1, and that of the circuit ID type is 0.

Figure 21 Sub-option 1 in normal padding format



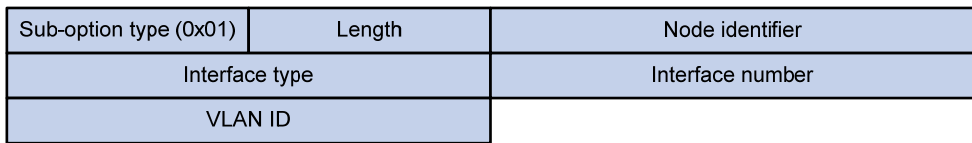
- **Sub-option 2**—Contains the MAC address of the DHCP relay agent interface or the MAC address of the DHCP snooping device that received the client's request. The value of the sub-option type is 2, and that of the remote ID type is 0.

Figure 22 Sub-option 2 in normal padding format



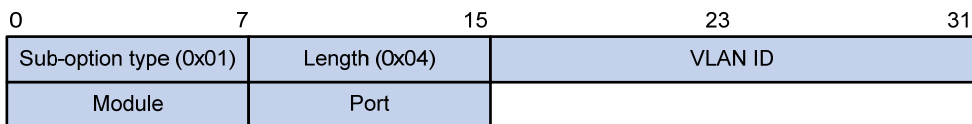
- Verbose padding format
 - **Sub-option 1**—Contains the user-specified access node identifier (ID of the device that adds Option 82 in DHCP messages), and the type, number, and VLAN ID of the interface that received the client's request. The VLAN ID field has a fixed length of 2 bytes. All the other padding contents of sub-option 1 are length variable. See [Figure 23](#).

Figure 23 Sub-option 1 in verbose padding format



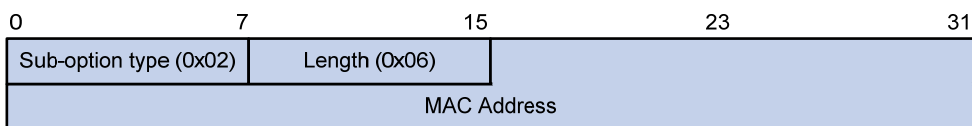
- **Sub-option 2**—Contains the MAC address of the DHCP relay agent interface or the MAC address of the DHCP snooping device that received the client's request. It has the same format as that in normal padding format. See [Figure 22](#).
- Private padding format
 - **Sub-option 1**—Contains the VLAN ID of the interface that received the client's request, module (subcard number of the receiving port) and port (port number of the receiving port). The value of the sub-option type is 1.

Figure 24 Sub-option 1 in private padding format



- **Sub-option 2**—Contains the MAC address of the DHCP snooping device that received the client's request. The value of the sub-option type is 2.

Figure 25 Sub-option 2 in private padding format



- **Sub-option 9**—Contains the Sysname and the primary IP address of the Loopback0 interface. The value of the sub-option type is 9.

Figure 26 Sub-option 9 in private padding format

0	7	15	23	31	
Sub-option type (0x09)		Length		Enterprise Number	
Enterprise Number			Information Length		Index (0x01)
Index (0x00)		Index (0x02)		Length	
Sysname			Index (0x03)		Index (0x04)
LoopBack0 IP					

- Standard padding format
 - **Sub-option 1**—Contains the VLAN ID of the interface that received the client's request, module (subcard number of the receiving port) and port (port number of the receiving port). The value of the sub-option type is 1, and the value of the circuit ID type is 0.

Figure 27 Sub-option 1 in standard padding format

0	7	15	23	31	
Sub-option type (0x01)		Length (0x06)		Circuit ID type (0x00)	
VLAN ID			Module		Port

- **Sub-option 2**—Contains the MAC address of the DHCP snooping device that received the client's request. It has the same format as that in normal padding format. See [Figure 22](#).

Option 184

Option 184 is a reserved option, and parameters in the option can be defined as needed. The device supports Option 184 carrying voice related parameters, so a DHCP client with voice functions can get an IP address along with specified voice parameters from the DHCP server.

Option 184 involves the following sub-options:

- **Sub-option 1**—IP address of the primary network calling processor, which serves as the network calling control source and provides program downloads.
- **Sub-option 2**—IP address of the backup network calling processor. DHCP clients contact the backup when the primary is unreachable.
- **Sub-option 3**—Voice VLAN ID and the result whether or not DHCP clients take this ID as the voice VLAN.
- **Sub-option 4**—Failover route that specifies the destination IP address and the called number. A Session Initiation Protocol (SIP) user uses this IP address and number to reach another SIP user when both the primary and backup calling processors are unreachable.

You must define sub-option 1 to make other sub-options take effect.

Protocols and standards

- RFC 2131, *Dynamic Host Configuration Protocol*
- RFC 2132, *DHCP Options and BOOTP Vendor Extensions*
- RFC 1542, *Clarifications and Extensions for the Bootstrap Protocol*
- RFC 3046, *DHCP Relay Agent Information Option*

- RFC 3442, *The Classless Static Route Option for Dynamic Host Configuration Protocol (DHCP) version 4*

Configuring DHCP server

Overview

The DHCP server is well suited to networks where:

- Manual configuration and centralized management are difficult to implement.
- Many hosts need to acquire IP addresses dynamically. This may be because the number of hosts exceeds the number of assignable IP addresses, so it is impossible to assign a fixed IP address to each host. For example, an ISP has a limited number of host addresses.
- A few hosts need fixed IP addresses.

DHCP address pool

Address pool types

DHCP address pools include common and extended address pools.

- **Common address pool**—Supports both static binding and dynamic allocation.
- **Extended address pool**—Supports only dynamic allocation.

Common address pool structure

The common address pool database is organized as a tree. The root of the tree is the address pool for natural networks, branches are address pools for subnets, and leaves are addresses statically bound to clients. For the same level address pools, a previously configured pool has a higher selection priority than a new one.

At the very beginning, subnets inherit network parameters and clients inherit subnet parameters. Therefore, common parameters, for example a DNS server address, should be configured at the highest (network or subnet) level of the tree. IP address lease durations are not inherited.

The new configuration at the higher level (parent) of the tree will be:

- Inherited if the lower level (child) has no such configuration.
- Overridden if the lower level (child) has such configuration.

NOTE:

The extended address pools on a DHCP server are independent of each other and no inheritance relationship exists among them.

Principles for selecting an address pool

The DHCP server observes the following principles to select an address pool when assigning an IP address to a client:

1. If there is an address pool where an IP address is statically bound to the MAC address or ID of the client, the DHCP server will select this address pool and assign the statically bound IP address to the client. For the configuration of this address pool, see "[Configuring static address allocation](#)."
2. If the receiving interface has an extended address pool referenced, the DHCP server will assign an IP address from this address pool. If no IP address is available in the address pool, the DHCP

server will fail to assign an address to the client. For the configuration of such an address pool, see "[Configuring dynamic address allocation for an extended address pool.](#)"

3. Otherwise, the DHCP server will select the smallest common address pool that contains the IP address of the receiving interface (if the client and the server reside on the same subnet), or the smallest common address pool that contains the IP address specified in the giaddr field of the client's request (if a DHCP relay agent is in-between). If no IP address is available in the address pool, the DHCP server will fail to assign an address to the client because it cannot assign an IP address from the parent address pool to the client. For the configuration of such an address pool, see "[Configuring dynamic address allocation.](#)"

For example, two common address pools, 1.1.1.0/24 and 1.1.1.0/25, are configured on the DHCP server. If the IP address of the interface receiving DHCP requests is 1.1.1.1/25, the DHCP server will select IP addresses for clients from address pool 1.1.1.0/25. If no IP address is available in the address pool, the DHCP server will fail to assign addresses to clients. If the IP address of the interface receiving DHCP requests is 1.1.1.130/25, the DHCP server will select IP addresses for clients from the 1.1.1.0/24 address pool.

NOTE:

To avoid wrong IP address allocation, keep the IP addresses for dynamic allocation within the subnet where the interface of the DHCP server or DHCP relay agent resides.

IP address allocation sequence

A DHCP server assigns an IP address to a client according to the following sequence:

1. The IP address statically bound to the client's MAC address or ID.
2. The IP address that was ever assigned to the client.
3. The IP address designated by the Option 50 field in a DHCP-DISCOVER message. Option 50 is the requested IP address field in DHCP-DISCOVER messages. It is padded by the client to specify the IP address that the client wants to obtain. The contents to be padded depend on the client.
4. The first assignable IP address found in an extended or common address pool.
5. The IP address that was a conflict or passed its lease duration.

If no IP address is assignable, the server will not respond.

DHCP server configuration task list

Task	Remarks
Configuring an address pool for the DHCP server	Required.
Enabling DHCP	Required.
Enabling the DHCP server on an interface	Required.
Applying an extended address pool on an interface	Required by the extended address pool configuration. When configuring a common address pool, ignore this task.
Configuring the DHCP server security functions	Optional.
Enabling client offline detection	Optional.

Task	Remarks
Enabling handling of Option 82	Optional.
Specifying a server's IP address for the DHCP client	Optional.
Specifying the threshold for sending trap messages	Optional.
Setting the DSCP value for DHCP packets	Optional.

Configuring an address pool for the DHCP server

Configuration task list

Task	Remarks			
Creating a DHCP address pool	Required.			
Configuring address allocation mode for a common address pool	<table border="0"> <tr> <td style="border-right: 1px solid black; padding-right: 10px;">Configuring static address allocation</td> <td rowspan="2">Required to configure either of the two for the common address pool configuration.</td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 10px;">Configuring dynamic address allocation</td> </tr> </table>	Configuring static address allocation	Required to configure either of the two for the common address pool configuration.	Configuring dynamic address allocation
Configuring static address allocation	Required to configure either of the two for the common address pool configuration.			
Configuring dynamic address allocation				
Configuring dynamic address allocation for an extended address pool	Required for the extended address pool configuration.			
Configuring a domain name suffix for the client				
Configuring DNS servers for the client				
Configuring WINS servers and NetBIOS node type for the client				
Configuring BIMS server information for the client				
Configuring gateways for the client	Optional.			
Configuring Option 184 parameters for the client with voice service				
Configuring the TFTP server and bootfile name for the client				
Specifying a server's IP address for the DHCP client				
Configuring self-defined DHCP options				

Creating a DHCP address pool

When creating a DHCP address pool, specify it as a common address pool or an extended address pool.

A common address pool and an extended address pool are different in address allocation mode configuration. Configurations of other parameters (such as the domain name suffix and DNS server address) for them are the same.

To create a DHCP address pool:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create a DHCP address pool and enter its view.	dhcp server ip-pool <i>pool-name</i> [extended]	No DHCP address pool is created by default.

Configuring address allocation mode for a common address pool

⚠ IMPORTANT:

You can configure either a static binding or dynamic address allocation for a common address pool, but not both.

You need to specify a subnet for dynamic address allocation. A static binding is a special address pool containing only one IP address.

Configuring static address allocation

Some DHCP clients, such as a WWW server, need fixed IP addresses. To provide a fixed IP address, you can create a static binding of a client's MAC address or client ID to an IP address in the DHCP address pool. A static binding is a special address pool containing only one IP address.

When the client with that MAC address or client ID requests an IP address, the DHCP server will assign the IP address from the binding to the client.

Follow these guidelines when you configure a static binding in a common address pool:

- Use the **static-bind ip-address** command together with **static-bind mac-address** or **static-bind client-identifier** to accomplish a static binding configuration.
- In a DHCP address pool, if you execute the **static-bind mac-address** command before the **static-bind client-identifier** command, the latter will overwrite the former and vice versa.
- If you use the **static-bind ip-address**, **static-bind mac-address**, or **static-bind client-identifier** command repeatedly in the DHCP address pool, the new configuration will overwrite the previous one.
- The IP address of the static binding cannot be an interface address of the DHCP server. Otherwise, an IP address conflict may occur and the bound client cannot obtain an IP address correctly.
- The ID of the static binding must be identical to the ID displayed by using the **display dhcp client verbose** command on the client. Otherwise, the client cannot obtain an IP address.
- The lease duration can be specified and takes effect for a static binding, but the lease duration from the **display dhcp server ip-in-use all** command output is still Unlimited.
- When the device serves as a DHCP client or BOOTP client, you must bind the DHCP client's ID to an IP address, or bind the BOOTP client's MAC address to an IP address on the DHCP server. Otherwise, the DHCP or BOOTP client cannot obtain a static IP address.
- If the interfaces on a DHCP client share the same MAC address, you must specify the client ID, rather than MAC address, in a static binding to identify the requesting interface. Otherwise, the client may fail to obtain an IP address.

To configure a static binding in a common address pool:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter common address pool view.	dhcp server ip-pool <i>pool-name</i>	N/A

Step	Command	Remarks
3. Specify the IP address.	static-bind ip-address <i>ip-address</i> [<i>mask-length</i> mask <i>mask</i>]	No IP addresses are statically bound by default.
4. Specify the MAC address or client ID.	<ul style="list-style-type: none"> Specify the MAC address: static-bind mac-address <i>mac-address</i> Specify the client ID: static-bind client-identifier <i>client-identifier</i> 	Use at least one command. Neither is bound statically by default.
5. Specify the lease duration for the IP address.	expired { day <i>day</i> [hour <i>hour</i> [minute <i>minute</i> [second <i>second</i>]]] unlimited }	Optional. By default, the lease duration of the IP address is unlimited.

Configuring dynamic address allocation

For dynamic address allocation, you must configure a DHCP address pool. For each address pool, you must specify one and only one address range, and the lease duration. A DHCP address pool can have only one lease duration.

To avoid address conflicts, configure the DHCP server to exclude IP addresses used by the gateway or FTP server from dynamic allocation.

Follow these guidelines when you configure dynamic address allocation for a common address pool:

- In common address pool view, using the **network** or **network ip range** command repeatedly overwrites the previous configuration.
- After you exclude IP addresses from automatic allocation by using the **dhcp server forbidden-ip** command, neither a common address pool nor an extended address pool can assign these IP addresses through dynamic address allocation.
- Using the **dhcp server forbidden-ip** command repeatedly can exclude multiple IP address ranges from allocation.

To configure dynamic address allocation for a common address pool:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter common address pool view.	dhcp server ip-pool <i>pool-name</i>	N/A
3. Specify a subnet.	network <i>network-address</i> [<i>mask-length</i> mask <i>mask</i>]	Not specified by default.
4. Specify the IP address range on the subnet for dynamic allocation.	network ip range <i>min-address</i> <i>max-address</i>	Optional. Not specified by default.
5. Specify the address lease duration.	expired { day <i>day</i> [hour <i>hour</i> [minute <i>minute</i>] [second <i>second</i>]] unlimited }	Optional. One day by default.
6. Return to system view.	quit	N/A

Step	Command	Remarks
7.	Exclude IP addresses from automatic allocation.	Optional. Except IP addresses of the DHCP server interfaces, all addresses in the DHCP address pool are assignable by default.

Configuring dynamic address allocation for an extended address pool

After the assignable IP address range and the mask are specified, the address pool becomes valid.

Extended address pools support dynamic address allocation only. Excluded IP addresses specified with the **forbidden-ip** command in DHCP address pool view are not assignable in the current extended address pool, but are assignable in other address pools.

To configure dynamic address allocation for an extended address pool:

Step	Command	Remarks
1.	Enter system view.	N/A
2.	Enter extended address pool view.	N/A
3.	Specify the IP address range.	Not specified by default.
4.	Specify the IP address mask.	Not specified by default.
5.	Specify the IP address range for the DHCP clients of a specific vendor.	Optional. Not configured by default.
6.	Specify the address lease duration.	Optional. One day by default.
7.	Exclude IP addresses from dynamic allocation.	Optional. Except IP addresses of the DHCP server interfaces, all addresses in the DHCP address pool are assignable by default.

Configuring a domain name suffix for the client

You can specify a domain name suffix in each DHCP address pool on the DHCP server to provide the clients with the domain name suffix. With this suffix assigned, the client only needs to input part of a domain name, and the system will add the domain name suffix for name resolution. For more information about DNS, see "Configuring IPv4 DNS."

To configure a domain name suffix in the DHCP address pool:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter DHCP address pool view.	dhcp server ip-pool <i>pool-name</i> [extended]	N/A
3. Specify a domain name suffix.	domain-name <i>domain-name</i>	Not specified by default

Configuring DNS servers for the client

A DHCP client contacts a Domain Name System (DNS) server to resolve names. You can specify up to eight DNS servers in the DHCP address pool.

To configure DNS servers in the DHCP address pool:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter DHCP address pool view.	dhcp server ip-pool <i>pool-name</i> [extended]	N/A
3. Specify DNS servers.	dns-list <i>ip-address&<1-8></i>	Not specified by default

Configuring WINS servers and NetBIOS node type for the client

A Microsoft DHCP client using NetBIOS protocol contacts a Windows Internet Naming Service (WINS) server for name resolution. Therefore, the DHCP server should assign a WINS server address when assigning an IP address to the client.

You can specify up to eight WINS servers in a DHCP address pool.

You must also specify a NetBIOS node type in a DHCP address pool. There are four NetBIOS node types:

- **b (broadcast)-node**—A b-node client sends the destination name in a broadcast message. The destination returns its IP address to the client after receiving the message.
- **p (peer-to-peer)-node**—A p-node client sends the destination name in a unicast message to the WINS server, and the WINS server returns the destination IP address.
- **m (mixed)-node**—An m-node client broadcasts the destination name. If it receives no response, it unicasts the destination name to the WINS server to get the destination IP address.
- **h (hybrid)-node**—An h-node client unicasts the destination name to the WINS server. If it receives no response, it broadcasts the destination name to get the destination IP address.

To configure WINS servers and NetBIOS node type in the DHCP address pool:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter DHCP address pool view.	dhcp server ip-pool <i>pool-name</i> [extended]	N/A

Step	Command	Remarks
3. Specify WINS server IP addresses.	nbns-list <i>ip-address</i> &<1-8>	Optional for b-node. No address is specified by default.
4. Specify the NetBIOS node type.	netbios-type { b-node h-node m-node p-node }	Not specified by default.

Configuring BIMS server information for the client

The DHCP server must provide DHCP clients with the branch intelligent management system (BIMS) server IP address, port number, shared key from the DHCP address pool, to enable DHCP clients to perform regular software update and backup by using configuration files obtained from a BIMS server.

To configure the BIMS server IP address, port number, and shared key in the DHCP address pool:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter DHCP address pool view.	dhcp server ip-pool <i>pool-name</i> [extended]	N/A
3. Specify the BIMS server IP address, port number, and shared key.	bims-server ip <i>ip-address</i> [port <i>port-number</i>] sharekey [cipher simple] <i>key</i>	Not specified by default

Configuring gateways for the client

You can specify up to eight gateways in a DHCP address pool.

To configure the gateways in the DHCP address pool:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter DHCP address pool view.	dhcp server ip-pool <i>pool-name</i> [extended]	N/A
3. Specify gateways.	gateway-list <i>ip-address</i> &<1-8>	No gateway is specified by default.

Configuring Option 184 parameters for the client with voice service

To assign voice calling parameters along with an IP address to DHCP clients with voice service, you must configure Option 184 on the DHCP server. For more information about Option 184, see "[DHCP overview](#)."

To configure option 184 parameters in the DHCP address pool:

Step	Command	Remarks
1. Enter system view.	system-view	N/A

Step	Command	Remarks
2. Enter DHCP address pool view.	dhcp server ip-pool <i>pool-name</i> [extended]	N/A
3. Specify the IP address of the primary network calling processor.	voice-config ncp-ip <i>ip-address</i>	Not specified by default. After you configure this command, the other Option 184 parameters take effect.
4. Specify the IP address of the backup network calling processor.	voice-config as-ip <i>ip-address</i>	Optional. Not specified by default.
5. Configure the voice VLAN.	voice-config voice-vlan <i>vlan-id</i> { disable enable }	Optional. Not configured by default.
6. Specify the failover IP address and dialer string.	voice-config fail-over <i>ip-address dialer-string</i>	Optional. No failover IP address or dialer string is specified by default.

Configuring the TFTP server and bootfile name for the client

For the DHCP server to support client auto-configuration, you must specify the IP address or name of a TFTP server and the bootfile name in the DHCP address pool. You do not need to perform any configuration on the DHCP client.

The DHCP client uses these parameters to contact the TFTP server and request the configuration file used for system initialization.

1. When a switch starts up without loading any configuration file, the system sets an active interface (such as the interface of the default VLAN) as the DHCP client to request from the DHCP server for parameters, such as an IP address and name of a TFTP server, and the bootfile name.
2. After getting related parameters, the DHCP client will send a TFTP request to obtain the configuration file from the specified TFTP server for system initialization. If the client cannot get such parameters, it will perform system initialization without loading any configuration file.

To configure the IP address and name of the TFTP server and the bootfile name in the DHCP address pool:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter DHCP address pool view.	dhcp server ip-pool <i>pool-name</i> [extended]	N/A
3. Specify the IP address or name of the TFTP server.	<ul style="list-style-type: none"> Specify the TFTP server: tftp-server ip-address <i>ip-address</i> Specify the name of the TFTP server: tftp-server domain-name <i>domain-name</i> 	Use either command. Not specified by default.
4. Specify the bootfile name.	bootfile-name <i>bootfile-name</i>	Not specified by default.

Specifying a server's IP address for the DHCP client

Some DHCP clients need to obtain configuration information from a server, such as a TFTP server. You can specify the IP address of that server in each address pool of the DHCP server. The DHCP server sends the server's IP address to DHCP clients along with other configuration information.

To specify the IP address of a server:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter DHCP address pool view.	dhcp server ip-pool <i>pool-name</i> [extended]	N/A
3. Specify the IP address of a server.	next-server <i>ip-address</i>	Not specified by default

Configuring self-defined DHCP options

⚠ CAUTION:

Be cautious when configuring self-defined DHCP options because such configuration may affect the DHCP operation process.

By configuring self-defined DHCP options, you can

- Define new DHCP options. New configuration options will come out with DHCP development. To support these new options, you can add them into the attribute list of the DHCP server.
- Define existing DHCP options. Vendors use Option 43 to define options that have no unified definitions in RFC 2132. The self-defined DHCP option enables DHCP clients to obtain vendor-specific information.
- Extend existing DHCP options. When the current DHCP options cannot meet the customers' requirements (for example, you cannot use the **dns-list** command to configure more than eight DNS server addresses), you can configure a self-defined option for extension.

To configure a self-defined DHCP option in the DHCP address pool:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter DHCP address pool view.	dhcp server ip-pool <i>pool-name</i> [extended]	N/A
3. Configure a self-defined DHCP option.	option code { ascii <i>ascii-string</i> hex <i>hex-string</i> <1-16> ip-address <i>ip-address</i> <1-8> }	No DHCP option is configured by default.

Table 2 Description of common options

Option	Option name	Corresponding command	Command parameter
3	Router Option	gateway-list	ip-address
6	Domain Name Server Option	dns-list	ip-address

Option	Option name	Corresponding command	Command parameter
15	Domain Name	domain-name	ascii
44	NetBIOS over TCP/IP Name Server Option	nbns-list	ip-address
46	NetBIOS over TCP/IP Node Type Option	netbios-type	hex
66	TFTP server name	tftp-server	ascii
67	Bootfile name	bootfile-name	ascii
43	Vendor Specific Information	N/A	hex

Enabling DHCP

Enable DHCP before performing other configurations.

To enable DHCP:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable DHCP.	dhcp enable	Disabled by default

Enabling the DHCP server on an interface

With the DHCP server enabled on an interface, upon receiving a client's request, the DHCP server will assign an IP address from its address pool to the DHCP client.

Configuration guidelines

Follow these guidelines when you enable the DHCP server on an interface:

- If a DHCP relay agent exists between the DHCP server and client, the DHCP server, regardless of whether the **subaddress** keyword is used, selects an IP address from the address pool containing the primary IP address of the DHCP relay agent's interface (connected to the client) for a requesting client.
- When the DHCP server and client communicate without Layer 3 forwarding:
 - With the keyword **subaddress** specified, the DHCP server will preferably assign an IP address from an address pool that resides on the same subnet as the primary IP address of the server interface (connecting to the client). If the address pool contains no assignable IP address, the server assigns an IP address from an address pool that resides on the same subnet as the secondary IP addresses of the server interface. If the interface has multiple secondary IP addresses, each address pool is tried in turn for address allocation. If the interface has no secondary IP addresses, the server is unable to assign an IP address to the client.
 - Without the keyword **subaddress** specified, the DHCP server can only assign an IP address from the address pool that resides on the same subnet as the primary IP address of the server interface.

Configuration procedure

To enable the DHCP server on an interface:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Enable the DHCP server on an interface.	dhcp select server global-pool [<i>subaddress</i>]	Optional. Enabled by default.

Applying an extended address pool on an interface

After you create an extended address pool and apply it on an interface, the DHCP server, upon receiving a client's request on the interface, attempts to assign the client the statically bound IP address first and then an IP address from the specified address pool. If no IP address is available in this address pool, address allocation fails, and the DHCP server will not assign the client any IP address from other address pools.

Only an extended address pool can be applied on the interface. The address pool to be referenced must already exist.

To apply an extended address pool on an interface:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Apply an extended address pool on the interface.	dhcp server apply ip-pool <i>pool-name</i>	Optional. By default, the DHCP server has no extended address pool applied on its interface, and assigns an IP address from a common address pool to a requesting client.

Configuring the DHCP server security functions

Configuration prerequisites

Before you configure the DHCP server security functions, complete the following tasks on the DHCP server:

1. Enable DHCP.
2. Configure the DHCP address pool.

Enabling unauthorized DHCP server detection

Unauthorized DHCP servers on a network may assign wrong IP addresses to DHCP clients.

With unauthorized DHCP server detection enabled, the DHCP server checks whether a DHCP request contains Option 54 (Server Identifier Option). If yes, the DHCP server records the IP address of each detected DHCP server that assigned an IP address to a requesting DHCP client in the option, and records the receiving interface. The administrator can use this information to check for unauthorized DHCP servers.

With the unauthorized DHCP server detection enabled, the switch logs each detected DHCP server once. The administrator can use the log information to find unauthorized DHCP servers.

To enable unauthorized DHCP server detection:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable unauthorized DHCP server detection.	dhcp server detect	Disabled by default

Configuring IP address conflict detection

With IP address conflict detection enabled, before assigning an IP address, the DHCP server pings that IP address by using ICMP. If the server receives a response within the specified period, it selects and pings another IP address. If it receives no response, the server continues to ping the IP address until the specified number of ping packets are sent. If still no response is received, the server assigns the IP address to the requesting client. (The DHCP client probes the IP address by sending gratuitous ARP packets.)

To configure IP address conflict detection:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Specify the number of ping packets.	dhcp server ping packets <i>number</i>	Optional. One ping packet by default. The value 0 indicates that no ping operation is performed.
3. Configure a timeout waiting for ping responses.	dhcp server ping timeout <i>milliseconds</i>	Optional. 500 ms by default. The value 0 indicates that no ping operation is performed.

Enabling client offline detection

With this feature enabled, the DHCP server considers a DHCP client goes offline when the ARP entry for the client ages out. In addition, it removes the client's IP-to-MAC binding entry.

Removing an ARP entry manually does not remove the corresponding client's IP-to-MAC binding.

To enable offline detection:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Enable offline detection.	dhcp server client-detect enable	Disabled by default

Enabling handling of Option 82

With Option 82 handling enabled, when the DHCP server receives a request with Option 82, it adds Option 82 into the response.

If the server is configured to ignore Option 82, it will assign an IP address to the client without adding Option 82 in the response message.

Configuration prerequisites

Before you enable Option 82 handling, complete the following tasks:

- **Configure the DHCP server**—Enable DHCP and configure the DHCP address pool.
- **Configure the relay agent or the device enabled with DHCP snooping**—For more information, see "[Configuring DHCP relay agent](#)" and "[Configuring DHCP snooping](#)."

Enabling Option 82 handling

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable the server to handle Option 82.	dhcp server relay information enable	Optional. Enabled by default.

Specifying the threshold for sending trap messages

Configuration prerequisites

Before you perform the configuration, use the **snmp-agent target-host** command to specify the destination address of the trap messages. For more information about the command, see *Network Management and Monitoring Command Reference*.

Configuration procedure

A DHCP server sends trap messages to the network management server when one of the following items reaches the specified threshold:

- The ratio of successfully allocated IP addresses to received DHCP requests
- The average IP address utilization of the address pool
- The maximum IP address utilization of the address pool

Trap messages help network administrators know the latest usage information about the DHCP server. To specify the threshold for sending trap messages:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Specify the threshold for sending trap messages to the network management server.	dhcp server threshold { allocated-ip <i>threshold-value</i> average-ip-use <i>threshold-value</i> max-ip-use <i>threshold-value</i> }	Optional. Disabled by default.

Setting the DSCP value for DHCP packets

An IPv4 packet header contains an 8-bit Type of Service (ToS) field. As defined in RFC 2474, the first six bits set the Differentiated Services Code Point (DSCP) value, and the last two bits are reserved. Network devices use the DSCP value as a reference to determine the packet priority for transmission.

To set the DSCP value for DHCP packets:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Set the DSCP value for DHCP packets sent by the DHCP server.	dhcp dscp <i>dscp-value</i>	Optional. By default, the DSCP value is 56.

Displaying and maintaining the DHCP server

ⓘ IMPORTANT:

A restart of the DHCP server or execution of the **reset dhcp server ip-in-use** command deletes all lease information. The DHCP server denies any DHCP request for lease extension, and the client must request an IP address again.

Task	Command	Remarks
Display information about IP address conflicts.	display dhcp server conflict { all ip <i>ip-address</i> } [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display information about lease expiration.	display dhcp server expired { all ip <i>ip-address</i> pool [<i>pool-name</i>] } [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display information about assignable IP addresses.	display dhcp server free-ip [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display IP addresses excluded from automatic allocation in the DHCP address pool.	display dhcp server forbidden-ip [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display information about bindings.	display dhcp server ip-in-use { all ip <i>ip-address</i> pool [<i>pool-name</i>] } [{ begin exclude include } <i>regular-expression</i>]	Available in any view

Task	Command	Remarks
Display information about DHCP server statistics.	display dhcp server statistics [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display tree organization information about address pools.	display dhcp server tree { all pool [<i>pool-name</i>] } [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Clear information about IP address conflicts.	reset dhcp server conflict { all ip <i>ip-address</i> }	Available in user view
Clear information about dynamic bindings.	reset dhcp server ip-in-use { all ip <i>ip-address</i> pool [<i>pool-name</i>] }	Available in user view
Clear information about DHCP server statistics.	reset dhcp server statistics	Available in user view

DHCP server configuration examples

DHCP networking involves the following two types:

- The DHCP server and client are on the same subnet and exchange messages directly.
- The DHCP server and client are not on the same subnet and they communicate with each other via a DHCP relay agent.

The DHCP server configuration for the two types is the same.

Static IP address assignment configuration example

Network requirements

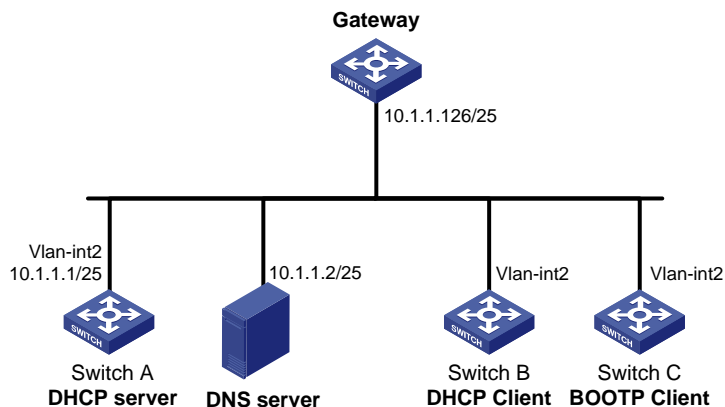
As shown in [Figure 28](#), Switch B (DHCP client) and Switch C (BOOTP client) obtain the static IP address, DNS server address, and gateway address from Switch A (DHCP server).

The client ID of VLAN-interface 2 on Switch B is:

3030-3066-2e65-3234-392e-3830-3530-2d56-6c61-6e2d-696e-7465-7266-6163-6532.

The MAC address of VLAN-interface 2 on Switch C is 000f-e249-8050.

Figure 28 Network diagram



Configuration procedure

1. Configure the IP address of VLAN-interface 2 on Switch A.

```
<SwitchA> system-view
[SwitchA] interface vlan-interface 2
[SwitchA-Vlan-interface2] ip address 10.1.1.1 25
[SwitchA-Vlan-interface2] quit
```

2. Configure the DHCP server:

Enable DHCP.

```
[SwitchA] dhcp enable
```

Enable the DHCP server on VLAN-interface 2.

```
[SwitchA] interface vlan-interface 2
[SwitchA-Vlan-interface2] dhcp select server global-pool
[SwitchA-Vlan-interface2] quit
```

Create DHCP address pool 0, configure a static binding, DNS server and gateway in it.

```
[SwitchA] dhcp server ip-pool 0
[SwitchA-dhcp-pool-0] static-bind ip-address 10.1.1.5 25
[SwitchA-dhcp-pool-0] static-bind client-identifier
3030-3066-2e65-3234-392e-3830-3530-2d56-6c61-6e2d-696e-7465-7266-6163-6532
[SwitchA-dhcp-pool-0] dns-list 10.1.1.2
[SwitchA-dhcp-pool-0] gateway-list 10.1.1.126
[SwitchA-dhcp-pool-0] quit
```

Create DHCP address pool 1, configure a static binding, DNS server and gateway in it.

```
[SwitchA] dhcp server ip-pool 1
[SwitchA-dhcp-pool-1] static-bind ip-address 10.1.1.6 25
[SwitchA-dhcp-pool-1] static-bind mac-address 000f-e249-8050
[SwitchA-dhcp-pool-1] dns-list 10.1.1.2
[SwitchA-dhcp-pool-1] gateway-list 10.1.1.126
```

Verifying the configuration

After the preceding configuration is complete, Switch B can obtain IP address 10.1.1.5 and other network parameters, and Switch C can obtain IP address 10.1.1.6 and other network parameters from Switch A. You can use the **display dhcp server ip-in-use** command on the DHCP server to view the IP addresses assigned to the clients.

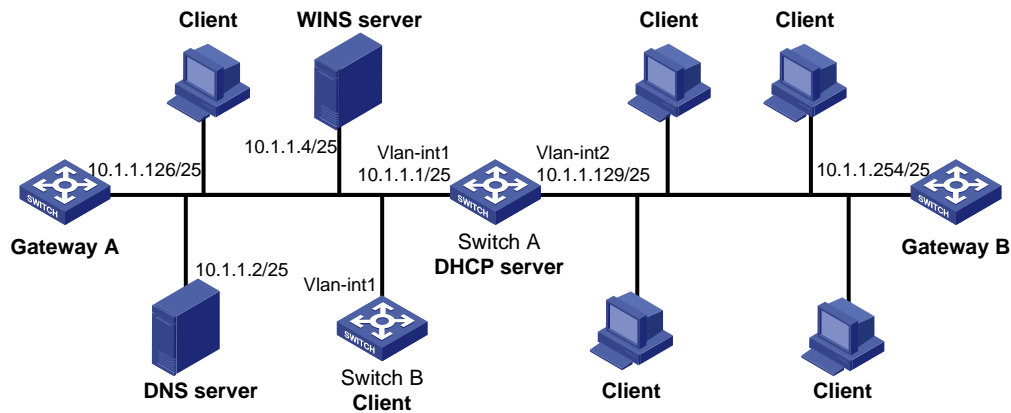
Dynamic IP address assignment configuration example

Network requirements

- As shown in [Figure 29](#), the DHCP server (Switch A) assigns IP addresses to clients in subnet 10.1.1.0/24, which is subnetted into 10.1.1.0/25 and 10.1.1.128/25.
- The IP addresses of VLAN-interfaces 1 and 2 on Switch A are 10.1.1.1/25 and 10.1.1.129/25 respectively.
- In address pool 10.1.1.0/25, configure the address lease duration as ten days and twelve hours, domain name suffix aabbcc.com, DNS server address 10.1.1.2/25, gateway 10.1.1.126/25, and WINS server 10.1.1.4/25.
- In address pool 10.1.1.128/25, configure the address lease duration as five days, domain name suffix aabbcc.com, DNS server address 10.1.1.2/25, and gateway address 10.1.1.254/25, and there is no WINS server address.

- The domain name and DNS server address on subnets 10.1.1.0/25 and 10.1.1.128/25 are the same. Therefore, the domain name suffix and DNS server address can be configured only for subnet 10.1.1.0/24. Subnet 10.1.1.128/25 can inherit the configuration of subnet 10.1.1.0/24.

Figure 29 Network diagram



Configuration procedure

1. Specify IP addresses for VLAN interfaces. (Details not shown.)
2. Configure the DHCP server:

Enable DHCP.

```
<SwitchA> system-view
[SwitchA] dhcp enable
```

Enable the DHCP server on VLAN-interface 1 and VLAN-interface 2.

```
[SwitchA] interface vlan-interface 1
[SwitchA-Vlan-interface1] dhcp select server global-pool
[SwitchA-Vlan-interface1] quit
[SwitchA] interface vlan-interface 2
[SwitchA-Vlan-interface2] dhcp select server global-pool
[SwitchA-Vlan-interface2] quit
```

Exclude IP addresses (addresses of the DNS server, WINS server and gateways).

```
[SwitchA] dhcp server forbidden-ip 10.1.1.2
[SwitchA] dhcp server forbidden-ip 10.1.1.4
[SwitchA] dhcp server forbidden-ip 10.1.1.126
[SwitchA] dhcp server forbidden-ip 10.1.1.254
```

Configure DHCP address pool 0 (subnet, client domain name suffix, and DNS server address).

```
[SwitchA] dhcp server ip-pool 0
[SwitchA-dhcp-pool-0] network 10.1.1.0 mask 255.255.255.0
[SwitchA-dhcp-pool-0] domain-name aabbcc.com
[SwitchA-dhcp-pool-0] dns-list 10.1.1.2
[SwitchA-dhcp-pool-0] quit
```

Configure DHCP address pool 1 (subnet, gateway, lease duration, and WINS server).

```
[SwitchA] dhcp server ip-pool 1
[SwitchA-dhcp-pool-1] network 10.1.1.0 mask 255.255.255.128
[SwitchA-dhcp-pool-1] gateway-list 10.1.1.126
[SwitchA-dhcp-pool-1] expired day 10 hour 12
```

```

[SwitchA-dhcp-pool-1] nbns-list 10.1.1.4
[SwitchA-dhcp-pool-1] quit
# Configure DHCP address pool 2 (subnet, gateway, and lease duration).
[SwitchA] dhcp server ip-pool 2
[SwitchA-dhcp-pool-2] network 10.1.1.128 mask 255.255.255.128
[SwitchA-dhcp-pool-2] expired day 5
[SwitchA-dhcp-pool-2] gateway-list 10.1.1.254

```

Verifying the configuration

After the preceding configuration is complete, clients on networks 10.1.1.0/25 and 10.1.1.128/25 can obtain IP addresses on the corresponding network and other network parameters from Switch A. You can use the **display dhcp server ip-in-use** command on the DHCP server to view the IP addresses assigned to the clients.

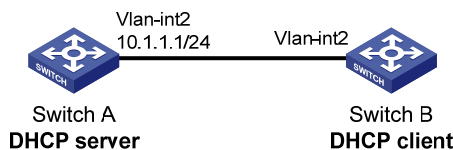
Self-defined option configuration example

Network requirements

As shown in [Figure 30](#), the DHCP client (Switch B) obtains an IP address and PXE server addresses from the DHCP server (Switch A). The IP address belongs to subnet 10.1.1.0/24. The PXE server addresses are 1.2.3.4 and 2.2.2.2.

The DHCP server assigns PXE server addresses to DHCP clients through Option 43, a self-defined option. The format of Option 43 and that of the PXE server address sub-option are shown in [Figure 18](#) and [Figure 20](#), respectively. The value of Option 43 configured on the DHCP server in this example is 80 0B 00 00 02 01 02 03 04 02 02 02 02. The number 80 is the value of the sub-option type. The number 0B is the value of the sub-option length. The numbers 00 00 are the value of the PXE server type. The number 02 indicates the number of servers. The numbers 01 02 03 04 02 02 02 02 indicate that the PXE server addresses are 1.2.3.4 and 2.2.2.2.

Figure 30 Network diagram



Configuration procedure

1. Specify IP addresses for the interfaces. (Details not shown.)
2. Configure the DHCP server:

```
# Enable DHCP.
```

```
<SwitchA> system-view
[SwitchA] dhcp enable
```

```
# Enable the DHCP server on VLAN-interface 2.
```

```
[SwitchA] interface vlan-interface 2
[SwitchA-Vlan-interface2] dhcp select server global-pool
[SwitchA-Vlan-interface2] quit
```

```
# Configure DHCP address pool 0.
```

```
[SwitchA] dhcp server ip-pool 0
[SwitchA-dhcp-pool-0] network 10.1.1.0 mask 255.255.255.0
```



```
[SwitchA-dhcp-pool-0] option 43 hex 80 0B 00 00 02 01 02 03 04 02 02 02 02
```

Verifying the configuration

After the preceding configuration is complete, Switch B can obtain its IP address on 10.1.1.0/24 and PXE server addresses from the Switch A. You can use the **display dhcp server ip-in-use** command on the DHCP server to view the IP addresses assigned to the clients.

Troubleshooting DHCP server configuration

Symptom

A client's IP address obtained from the DHCP server conflicts with another IP address.

Analysis

A host on the subnet may have the same IP address.

Solution

1. Disable the client's network adapter or disconnect the client's network cable. Ping the IP address of the client from another host to check whether there is a host using the same IP address.
2. If a ping response is received, the IP address has been manually configured on a host. Execute the **dhcp server forbidden-ip** command on the DHCP server to exclude the IP address from dynamic allocation.
3. Enable the network adapter or connect the network cable. Release the IP address and obtain another one on the client. For example, to release the IP address and obtain another one on a Windows XP DHCP client:
 - a. In a Windows environment, select **Start > Run**. Enter **cmd** in the dialog box, and click **OK** to enter the command line interface.
 - b. Enter **ipconfig/release** to relinquish the IP address.
 - c. Enter **ipconfig/renew** to obtain another IP address.

Configuring DHCP relay agent

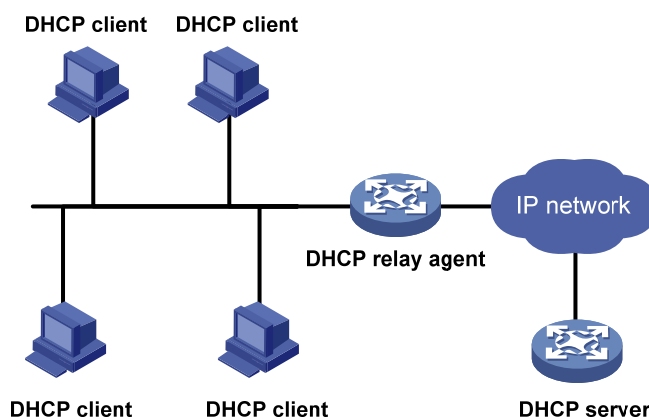
The DHCP relay agent configuration is supported only on VLAN interfaces.

Overview

Via a relay agent, DHCP clients can communicate with a DHCP server on another subnet to obtain configuration parameters. DHCP clients on different subnets can contact the same DHCP server rather than having a DHCP server on each subnet. This centralizes management and reduces cost reduction.

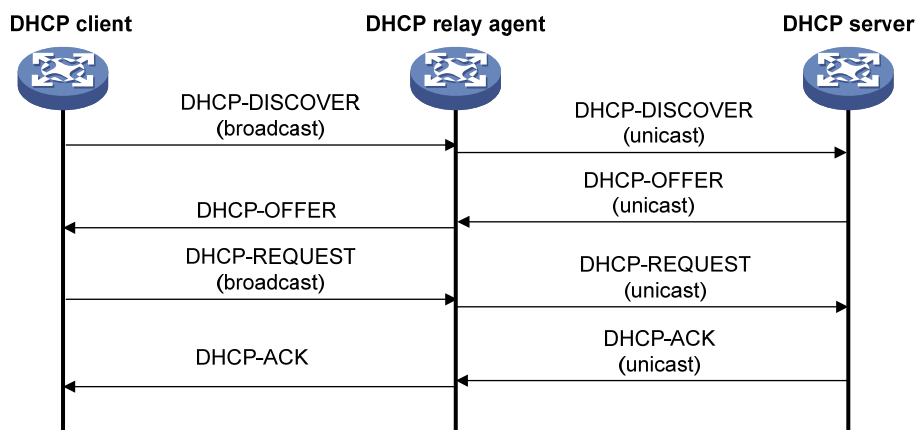
Fundamentals

Figure 31 DHCP relay agent application



The DHCP server and client interact with each other in the same way with or without a relay agent (see "DHCP overview").

Figure 32 DHCP relay agent work process



1. After receiving a DHCP-DISCOVER or DHCP-REQUEST broadcast message from a DHCP client, the DHCP relay agent fills the giaddr field of the message with its IP address and forwards the message to the designated DHCP server in unicast mode.

- Based on the giaddr field, the DHCP server returns an IP address and other configuration parameters to the relay agent, and the relay agent conveys them to the client.

DHCP relay agent support for Option 82

Option 82 records location information about the DHCP client, letting the administrator locate the DHCP client for security control and accounting purposes. For more information, see "[DHCP overview](#)."

If the DHCP relay agent supports Option 82, it handles a client's request according to the contents defined in Option 82, if any. The handling strategies are described in [Table 3](#).

If a reply returned by the DHCP server contains Option 82, the DHCP relay agent removes the Option 82 before forwarding the reply to the client.

Table 3 Handling strategies of the DHCP relay agent

If a client's requesting message has...	Handling strategy	Padding format	The DHCP relay agent will...
Option 82	Drop	Random	Drop the message.
	Keep	Random	Forward the message without changing Option 82.
		normal	Forward the message after replacing the original Option 82 with the Option 82 padded in normal format.
		verbose	Forward the message after replacing the original Option 82 with the Option 82 padded in verbose format.
	Replace	user-defined	Forward the message after replacing the original Option 82 with the user-defined Option 82.
	no Option 82	N/A	normal
N/A		verbose	Forward the message after adding the Option 82 padded in verbose format.
N/A		user-defined	Forward the message after adding the user-defined Option 82.

DHCP relay agent configuration task list

Task	Remarks
Enabling DHCP	Required
Enabling the DHCP relay agent on an interface	Required
Correlating a DHCP server group with a relay agent interface	Required
Configuring the DHCP relay agent security functions	Optional
Enabling offline detection	Optional
Configuring the DHCP relay agent to release an IP address	Optional

Task	Remarks
Configuring the DHCP relay agent to support Option 82	Optional
Setting the DSCP value for DHCP packets	Optional

Enabling DHCP

Enable DHCP before performing other configurations related to the DHCP relay agent.

To enable DHCP:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable DHCP.	dhcp enable	Disabled by default

Enabling the DHCP relay agent on an interface

With the DHCP relay agent enabled, an interface forwards incoming DHCP requests to a DHCP server for address allocation.

The IP address pool containing the IP address of the DHCP relay agent enabled interface must be configured on the DHCP server. Otherwise, the DHCP clients connected to the relay agent cannot obtain correct IP addresses.

To enable the DHCP relay agent on an interface:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Enable the DHCP relay agent on the current interface.	dhcp select relay	With DHCP enabled, interfaces operate in the DHCP server mode.

Correlating a DHCP server group with a relay agent interface

To improve reliability, you can specify several DHCP servers as a group on the DHCP relay agent and correlate a relay agent interface with the server group. When the interface receives request messages from clients, the relay agent will forward them to all the DHCP servers of the group.

Configuration guidelines

Follow these guidelines when you correlate a DHCP server group with a relay agent interface:

- You can specify up to twenty DHCP server groups on the relay agent.

- By executing the **dhcp relay server-group** command repeatedly, you can specify up to eight DHCP server addresses for each DHCP server group.
- The IP addresses of DHCP servers and those of relay agent's interfaces that connect DHCP clients cannot be on the same subnet. Otherwise, the client cannot obtain an IP address.
- A DHCP server group can correlate with one or multiple DHCP relay agent interfaces, while a relay agent interface can only correlate with one DHCP server group. Using the **dhcp relay server-select** command repeatedly overwrites the previous configuration. However, if the specified DHCP server group does not exist, the interface still uses the previous correlation.
- The *group-id* argument in the **dhcp relay server-select** command is configured by using the **dhcp relay server-group** command.

Configuration procedure

To correlate a DHCP server group with a relay agent interface:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create a DHCP server group and add a server into the group.	dhcp relay server-group <i>group-id</i> ip <i>ip-address</i>	Not created by default.
3. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
4. Correlate the DHCP server group with the current interface.	dhcp relay server-select <i>group-id</i>	By default, no interface is correlated with any DHCP server group.

Configuring the DHCP relay agent security functions

Configuring address check

Address check can block illegal hosts from accessing external networks.

With this feature enabled, the DHCP relay agent can dynamically record clients' IP-to-MAC bindings after they obtain IP addresses through DHCP. This feature also supports static bindings. You can also configure static IP-to-MAC bindings on the DHCP relay agent, so users can access external networks using fixed IP addresses.

Upon receiving a packet from a host, the DHCP relay agent checks the source IP and MAC addresses in the packet against the recorded dynamic and static bindings. If no match is found, the DHCP relay agent does not learn the ARP entry of the host, and will not forward any reply to the host, so the host cannot access external networks via the DHCP relay agent.

Configuration guidelines

Follow these guidelines when you create a static binding and enable address check:

- The **dhcp relay address-check enable** command can be executed only on VLAN interfaces.

- Before enabling address check on an interface, you must enable the DHCP service, and enable the DHCP relay agent on the interface. Otherwise, the address check configuration is ineffective.
- The **dhcp relay address-check enable** command only checks IP and MAC addresses but not interfaces.
- When using the **dhcp relay security static** command to bind an interface to a static binding entry, make sure that the interface is configured as a DHCP relay agent. Otherwise, address entry conflicts may occur.

Configuration procedure

To create a static binding and enable address check:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create a static binding.	dhcp relay security static <i>ip-address mac-address [interface interface-type interface-number]</i>	Optional. No static binding is created by default.
3. Enter interface view.	interface <i>interface-type interface-number</i>	N/A
4. Enable address check.	dhcp relay address-check enable	Disabled by default.

Configuring periodic refresh of dynamic client entries

A DHCP client unicasts a DHCP-RELEASE message to the DHCP server to release its IP address. The DHCP relay agent simply conveys the message to the DHCP server and does not remove the IP-to-MAC entry of the client.

When this feature is enabled, the DHCP relay agent uses the IP address of a client and the MAC address of the DHCP relay interface to send a DHCP-REQUEST message to the DHCP server at specified intervals.

- If the server returns a DHCP-ACK message or does not return any message within a specific interval, the DHCP relay agent ages out the entry.
- If the server returns a DHCP-NAK message, the relay agent keeps the entry.

To configure periodic refresh of dynamic client entries:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable periodic refresh of dynamic client entries.	dhcp relay security refresh enable	Optional. Enabled by default.
3. Configure the refresh interval.	dhcp relay security tracker { <i>interval</i> auto }	Optional. auto by default. (auto interval is calculated by the relay agent according to the number of client entries.)

Enabling unauthorized DHCP server detection

Unauthorized DHCP servers may assign wrong IP addresses to DHCP clients.

With unauthorized DHCP servers detection enabled, the DHCP relay agent checks whether a request contains Option 54 (Server Identifier Option). If yes, the DHCP relay agent records the IP address of each detected DHCP server that assigned an IP address to a requesting DHCP client in the option, and records the receiving interface. The administrator can use this information to check for unauthorized DHCP servers.

The relay agent logs a DHCP server only once.

To enable unauthorized DHCP server detection:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable unauthorized DHCP server detection.	dhcp relay server-detect	Disabled by default

Enabling DHCP starvation attack protection

A DHCP starvation attack occurs when an attacker constantly sends forged DHCP requests using different MAC addresses in the **chaddr** field to a DHCP server. This exhausts the IP address resources of the DHCP server so legitimate DHCP clients cannot obtain IP addresses. The DHCP server may also fail to work because of exhaustion of system resources.

- To relieve a DHCP starvation attack that uses DHCP packets encapsulated with different source MAC addresses, you can limit the number of ARP entries that a Layer 3 interface can learn or MAC addresses that a Layer 2 port can learn. You can also configure an interface that has learned the maximum MAC addresses to discard packets whose source MAC addresses are not in the MAC address table.
- To prevent a DHCP starvation attack that uses DHCP requests encapsulated with the same source MAC address, enable MAC address check on the DHCP relay agent. With this function enabled, the DHCP relay agent compares the **chaddr** field of a received DHCP request with the source MAC address field of the frame. If they are the same, the DHCP relay agent decides this request as valid and forwards it to the DHCP server. If not, it discards the DHCP request.

To enable MAC address check:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Enable MAC address check.	dhcp relay check mac-address	Disabled by default

NOTE:

DHCP relay agents change the source MAC addresses when forwarding DHCP packets. Therefore, you can enable MAC address check only on a DHCP relay agent directly connected to DHCP clients. Otherwise, valid DHCP packets may be discarded and clients cannot obtain IP addresses.

Enabling offline detection

The DHCP relay agent checks whether a user is online by learning the ARP entry. When an ARP entry is aged out, the corresponding client is considered to be offline.

With this function enabled on an interface, the DHCP relay agent removes a client's IP-to-MAC entry when it is aged out, and sends a DHCP-RELEASE message to the DHCP server to release the IP address of the client. Removing an ARP entry manually does not remove the corresponding client's IP-to-MAC binding. When the client goes offline, use the **undo dhcp relay security** command to remove the IP-to-MAC binding manually.

To enable offline detection:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Enable offline detection.	dhcp relay client-detect enable	Disabled by default

Configuring the DHCP relay agent to release an IP address

You can configure the relay agent to release a client's IP address. The relay agent sends a DHCP-RELEASE message that contains the IP address. Upon receiving the DHCP-RELEASE message, the DHCP server releases the IP address. Meanwhile, the client entry is removed from the DHCP relay agent. Dynamic client entries can be generated after you enable address check or IP source guard on the DHCP relay agent. For more information about IP source guard, see *Security Configuration Guide*.

To configure the DHCP relay agent to send DHCP-RELEASE messages:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Configure the DHCP relay agent to release an IP address.	dhcp relay release ip <i>client-ip</i>	The IP address must be in a dynamic client entry.

Configuring the DHCP relay agent to support Option 82

Configuration prerequisites

Before you perform this configuration, complete the following tasks:

1. Enable DHCP.
2. Enable the DHCP relay agent on the specified interface.
3. Correlate a DHCP server group with relay agent interfaces.

Configuration guidelines

- To support Option 82, perform related configuration on both the DHCP server and relay agent. See "Configuring DHCP server" for DHCP server configuration of this kind.
- If the handling strategy of the DHCP relay agent is configured as **replace**, you must configure a padding format for Option 82. If the handling strategy is **keep** or **drop**, you need not configure any padding format.
- If sub-option 1 (node identifier) of Option 82 is padded with the device name (sysname) of a node, the device name must contain no spaces. Otherwise, the DHCP relay agent will drop the message.

Configuration procedure

To configure the DHCP relay agent to support Option 82:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Enable the relay agent to support Option 82.	dhcp relay information enable	Disabled by default.
4. Configure the handling strategy for requesting messages containing Option 82.	dhcp relay information strategy { drop keep replace }	Optional. replace by default.
5. Configure non-user-defined Option 82.	<ul style="list-style-type: none"> • Configure the padding format for Option 82: dhcp relay information format { normal verbose [node-identifier { mac sysname user-defined <i>node-identifier</i> }] } • Configure the code type for the circuit ID sub-option: dhcp relay information circuit-id format-type { ascii hex } • Configure the code type for the remote ID sub-option: dhcp relay information remote-id format-type { ascii hex } 	Optional. By default: <ul style="list-style-type: none"> • The padding format for Option 82 is normal. • The code type for the circuit ID sub-option depends on the padding format of Option 82. Each field has its own code type. • The code type for the remote ID sub-option is hex. The code type configurations for the circuit ID sub-option and remote ID sub-option apply to non-user-defined Option 82 only.
6. Configure user-defined Option 82.	<ul style="list-style-type: none"> • Configure the padding content for the circuit ID sub-option: dhcp relay information circuit-id string <i>circuit-id</i> • Configure the padding content for the remote ID sub-option: dhcp relay information remote-id string { <i>remote-id</i> sysname } 	Optional. By default, the padding content depends on the padding format of Option 82.

Setting the DSCP value for DHCP packets

An IPv4 packet header contains an 8-bit Type of Service (ToS) field. As defined in RFC 2474, the first six bits set the Differentiated Services Code Point (DSCP) value, and the last two bits are reserved. Network devices use the DSCP value as a reference to determine the packet priority for transmission.

To set the DSCP value for DHCP packets:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Set the DSCP value for DHCP packets sent by the DHCP relay agent.	dhcp dscp <i>dscp-value</i>	Optional. By default, the DSCP value is 56.

Displaying and maintaining the DHCP relay agent

Task	Command	Remarks
Display information about DHCP server groups correlated to a specific interface or all interfaces.	display dhcp relay { all interface <i>interface-type interface-number</i> } [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display Option 82 configuration information on the DHCP relay agent.	display dhcp relay information { all interface <i>interface-type interface-number</i> } [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display information about bindings of DHCP relay agents.	display dhcp relay security [<i>ip-address</i> dynamic static] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display statistics about bindings of DHCP relay agents.	display dhcp relay security statistics [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display information about the refreshing interval for entries of dynamic IP-to-MAC bindings.	display dhcp relay security tracker [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display information about the configuration of a specific DHCP server group or all DHCP server groups.	display dhcp relay server-group { <i>group-id</i> all } [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display packet statistics on relay agent.	display dhcp relay statistics [server-group { <i>group-id</i> all }] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Clear packet statistics from relay agent.	reset dhcp relay statistics [server-group <i>group-id</i>]	Available in user view

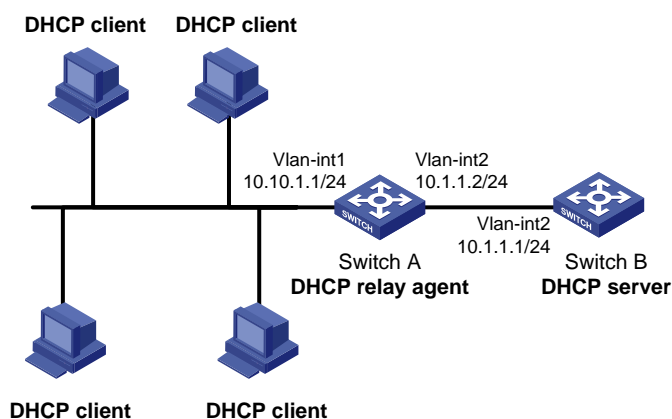
DHCP relay agent configuration examples

DHCP relay agent configuration example

Network requirements

As shown in [Figure 33](#), DHCP clients reside on network 10.10.1.0/24. The IP address of the DHCP server is 10.1.1.1/24. Because the DHCP clients reside on a different network than the DHCP server, a DHCP relay agent is deployed to forward messages between DHCP clients and the DHCP server. VLAN-interface 1 on the DHCP relay agent (Switch A) connects to the network where DHCP clients reside. The IP address of VLAN-interface 1 is 10.10.1.1/24 and the IP address of VLAN-interface 2 is 10.1.1.2/24.

Figure 33 Network diagram



Configuration procedure

The DHCP relay agent and server are on different subnets, so configure a static route or dynamic routing protocol to make them reachable to each other.

Configurations on the DHCP server are also required to guarantee the client-server communication via the DHCP relay agent. For DHCP server configuration information, see "[Configuring DHCP server.](#)"

Specify IP addresses for the interfaces. (Details not shown.)

Enable DHCP.

```
<SwitchA> system-view  
[SwitchA] dhcp enable
```

Add DHCP server 10.1.1.1 into DHCP server group 1.

```
[SwitchA] dhcp relay server-group 1 ip 10.1.1.1
```

Enable the DHCP relay agent on VLAN-interface 1.

```
[SwitchA] interface vlan-interface 1  
[SwitchA-Vlan-interface1] dhcp select relay
```

Correlate VLAN-interface 1 to DHCP server group 1.

```
[SwitchA-Vlan-interface1] dhcp relay server-select 1
```

After the preceding configuration is complete, DHCP clients can obtain IP addresses and other network parameters through the DHCP relay agent from the DHCP server. You can use the **display dhcp relay statistics** command to view statistics of DHCP packets forwarded by DHCP relay agents. After you enable

address check of the DHCP relay agents with the **dhcp relay address-check enable** command, use the **display dhcp relay security** command to view bindings of DHCP relay agents

DHCP relay agent Option 82 support configuration example

Network requirements

- As shown in [Figure 33](#), enable Option 82 on the DHCP relay agent (Switch A).
- Configure the handling strategy for DHCP requests containing Option 82 as **replace**.
- Configure the padding content for the circuit ID sub-option as **company001** and for the remote ID sub-option as **device001**.
- Switch A forwards DHCP requests to the DHCP server (Switch B) after replacing Option 82 in the requests, so that the DHCP clients can obtain IP addresses.

Configuration procedure

Configurations on the DHCP server are also required to make the Option 82 configurations function normally.

Specify IP addresses for the interfaces. (Details not shown.)

Enable DHCP.

```
<SwitchA> system-view  
[SwitchA] dhcp enable
```

Add DHCP server 10.1.1.1 into DHCP server group 1.

```
[SwitchA] dhcp relay server-group 1 ip 10.1.1.1
```

Enable the DHCP relay agent on VLAN-interface 1.

```
[SwitchA] interface vlan-interface 1  
[SwitchA-Vlan-interface1] dhcp select relay
```

Correlate VLAN-interface 1 to DHCP server group 1.

```
[SwitchA-Vlan-interface1] dhcp relay server-select 1
```

Enable the DHCP relay agent to support Option 82, and perform Option 82-related configurations.

```
[SwitchA-Vlan-interface1] dhcp relay information enable  
[SwitchA-Vlan-interface1] dhcp relay information strategy replace  
[SwitchA-Vlan-interface1] dhcp relay information circuit-id string company001  
[SwitchA-Vlan-interface1] dhcp relay information remote-id string device001
```

Troubleshooting DHCP relay agent configuration

Symptom

DHCP clients cannot obtain any configuration parameters via the DHCP relay agent.

Analysis

Problems may occur with the DHCP relay agent or server configuration.

Solution

To locate the problem, enable debugging and execute the **display** command on the DHCP relay agent to view the debugging information and interface state information.

Verify that:

- The DHCP is enabled on the DHCP server and relay agent.
- The address pool on the same subnet where DHCP clients reside is available on the DHCP server.
- The DHCP server and DHCP relay agent are reachable to each other.
- The relay agent interface connected to DHCP clients is correlated with a correct DHCP server group and the IP addresses of the group members are correct.

Configuring DHCP client

With DHCP client enabled, an interface uses DHCP to obtain configuration parameters such as an IP address from the DHCP server.

Configuration restrictions

- The DHCP client configuration is supported only on VLAN interfaces.
- When multiple VLAN interfaces with the same MAC address use DHCP for IP address acquisition via a relay agent, the DHCP server cannot be a Windows Server 2000 or Windows Server 2003.

Enabling the DHCP client on an interface

Follow these guidelines when you enable the DHCP client on an interface:

- An interface can be configured to acquire an IP address in multiple ways. The latest configuration overwrites the previous one.
- Secondary IP addresses cannot be configured on an interface that is enabled with the DHCP client.
- If the IP address that interface A obtains from the DHCP server is on the same network segment as the IP address of interface B, interface A neither uses the IP address nor requests any IP address from the DHCP server unless you do the following: Delete the IP address of interface B and bring up interface A again by first executing the **shutdown** command and then the **undo shutdown** command, or, re-enable the DHCP client on interface A by executing the **undo ip address dhcp-alloc** command and then the **ip address dhcp-alloc** command.

To enable the DHCP client on an interface:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Enable the DHCP client on the interface.	ip address dhcp-alloc [client-identifier mac <i>interface-type</i> <i>interface-number</i>]	<ul style="list-style-type: none">• If the device starts up with initial settings, it uses the software initial setting that an interface does not use DHCP for IP address acquisition.• If the device starts up with default configuration file, it uses the software default setting that an interface uses its MAC address to be the client ID for IP address acquisition. <p>For more information about initial settings and default configuration file, see <i>Fundamentals Configuration Guide</i>.</p>

Setting the DSCP value for DHCP packets

An IPv4 packet header contains an 8-bit Type of Service (ToS) field. As defined in RFC 2474, the first six bits set the Differentiated Services Code Point (DSCP) value, and the last two bits are reserved. Network devices use the DSCP value as a reference to determine the packet priority for transmission.

To set the DSCP value for DHCP packets:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Set the DSCP value for DHCP packets sent by the DHCP client.	dhcp client dscp <i>dscp-value</i>	Optional. By default, the DSCP value is 56.

Displaying and maintaining the DHCP client

Task	Command	Remarks
Display specified configuration information.	display dhcp client [<i>verbose</i>] [interface <i>interface-type interface-number</i>] [{ begin exclude include } <i>regular-expression</i>]	Available in any view

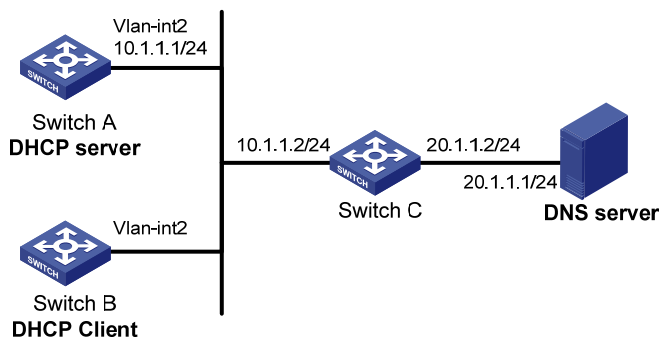
DHCP client configuration example

Network requirements

As shown in Figure 34, on a LAN, Switch B contacts the DHCP server via VLAN-interface 2 to obtain an IP address, DNS server address, and static route information. The DHCP client IP address resides on network 10.1.1.0/24. The DNS server address is 20.1.1.1. The next hop of the static route to network 20.1.1.0/24 is 10.1.1.2.

The DHCP server uses Option 121 to assign static route information to DHCP clients. The destination descriptor field comprises two parts, subnet mask length and destination network address. In this example, the value of the destination descriptor field takes 18 14 01 01, a hexadecimal number indicating that the subnet mask length is 24 and destination network address is 20.1.1.0. The value of the next hop address field takes 0A 01 01 02, a hexadecimal number indicating that the next hop is 10.1.1.2.

Figure 34 Network diagram



Configuration procedure

1. Configure Switch A:

Specify the IP address of VLAN-interface 2.

```
<SwitchA> system-view
[SwitchA] interface vlan-interface 2
[SwitchA-Vlan-interface2] ip address 10.1.1.1 24
[SwitchA-Vlan-interface2] quit
```

Enable the DHCP service.

```
[SwitchA] dhcp enable
```

Exclude an IP address from automatic allocation.

```
[SwitchA] dhcp server forbidden-ip 10.1.1.2
```

Configure DHCP address pool 0 and specify the subnet, lease duration, DNS server address, and a static route to subnet 20.1.1.0/24.

```
[SwitchA] dhcp server ip-pool 0
[SwitchA-dhcp-pool-0] network 10.1.1.0 mask 255.255.255.0
[SwitchA-dhcp-pool-0] expired day 10
[SwitchA-dhcp-pool-0] dns-list 20.1.1.1
[SwitchA-dhcp-pool-0] option 121 hex 18 14 01 01 0A 01 01 02
```

2. Enable the DHCP client on VLAN-interface 2 of Switch B.

```
<SwitchB> system-view
[SwitchB] interface vlan-interface 2
[SwitchB-Vlan-interface2] ip address dhcp-alloc
```

Verifying the configuration

Use the **display dhcp client** command to view the IP address and other network parameters assigned to Switch B.

```
[SwitchB-Vlan-interface2] display dhcp client verbose
Vlan-interface2 DHCP client information:
Current machine state: BOUND
Allocated IP: 10.1.1.3 255.255.255.0
Allocated lease: 864000 seconds, T1: 432000 seconds, T2: 756000 seconds
Lease from 2009.02.20 11:06:35 to 2009.03.02 11:06:35
DHCP server: 10.1.1.1
Transaction ID: 0x410090f0
Classless static route:
  Destination: 20.1.1.0, Mask: 255.255.255.0, NextHop: 10.1.1.2
DNS server: 20.1.1.1
Client ID: 3030-3066-2e65-3230-
          302e-3030-3032-2d45-
          7468-6572-6e65-7430-
          2f30
T1 will timeout in 4 days 23 hours 59 minutes 50 seconds.
```

Use the **display ip routing-table** command to view the route information on Switch B. A static route to network 20.1.1.0/24 is added to the routing table.

```
[SwitchB-Vlan-interface2] display ip routing-table
```


Routing Tables: Public

Destinations : 5 Routes : 5

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
10.1.1.0/24	Direct	0	0	10.1.1.3	Vlan2
10.1.1.3/32	Direct	0	0	127.0.0.1	InLoop0
20.1.1.0/24	Static	70	0	10.1.1.2	Vlan2
127.0.0.0/8	Direct	0	0	127.0.0.1	InLoop0
127.0.0.1/32	Direct	0	0	127.0.0.1	InLoop0

Configuring DHCP snooping

The DHCP snooping-enabled device must be either between the DHCP client and relay agent, or between the DHCP client and server. It does not work if it is between the DHCP relay agent and DHCP server.

DHCP snooping functions

DHCP snooping can:

1. Ensure that DHCP clients obtain IP addresses from authorized DHCP servers.
2. Record IP-to-MAC mappings of DHCP clients.

Ensuring that DHCP clients obtain IP addresses from authorized DHCP servers

With DHCP snooping, the ports of a switch can be configured as trusted or untrusted to make sure that clients obtain IP addresses only from authorized DHCP servers.

- **Trusted**—A trusted port forwards DHCP messages normally to ensure the clients get IP addresses from an authorized DHCP server.
- **Untrusted**—An untrusted port discards received DHCP-ACK and DHCP-OFFER messages to avoid IP address allocation from any unauthorized server.

Configure ports that connect to authorized DHCP servers or other DHCP snooping devices as trusted, and configure other ports as untrusted.

Recording IP-to-MAC mappings of DHCP clients

DHCP snooping reads DHCP-REQUEST messages and DHCP-ACK messages from trusted ports to record DHCP snooping entries. A DHCP snooping entry includes the MAC and IP addresses of the client, the port that connects to the DHCP client, and the VLAN of the port. Using DHCP snooping entries, DHCP snooping can implement the following functions:

- **ARP detection**—Whether ARP packets are sent from an authorized client is determined based on DHCP snooping entries. This feature prevents ARP attacks from unauthorized clients. For more information, see *Security Configuration Guide*.
- **IP source guard**—IP source guard uses dynamic binding entries generated by DHCP snooping to filter packets on a per-port basis. This prevents unauthorized packets from traveling through. For more information, see *Security Configuration Guide*.

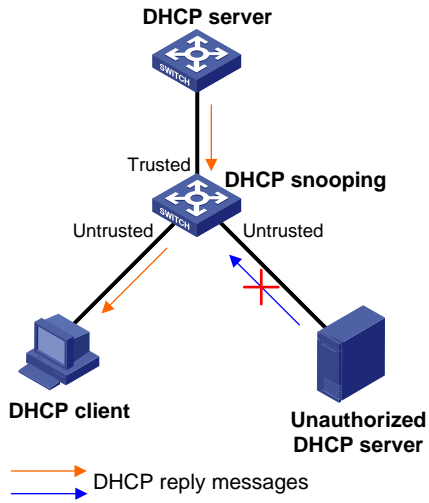
Application environment of trusted ports

Configuring a trusted port connected to a DHCP server

As shown in [Figure 35](#), the DHCP snooping device port that is connected to an authorized DHCP server should be configured as a trusted port. The trusted port forwards reply messages from the authorized

DHCP server to the client, but the untrusted port does not forward reply messages from the unauthorized DHCP server. This ensures that the DHCP client obtains an IP address from the authorized DHCP server.

Figure 35 Configuring trusted and untrusted ports

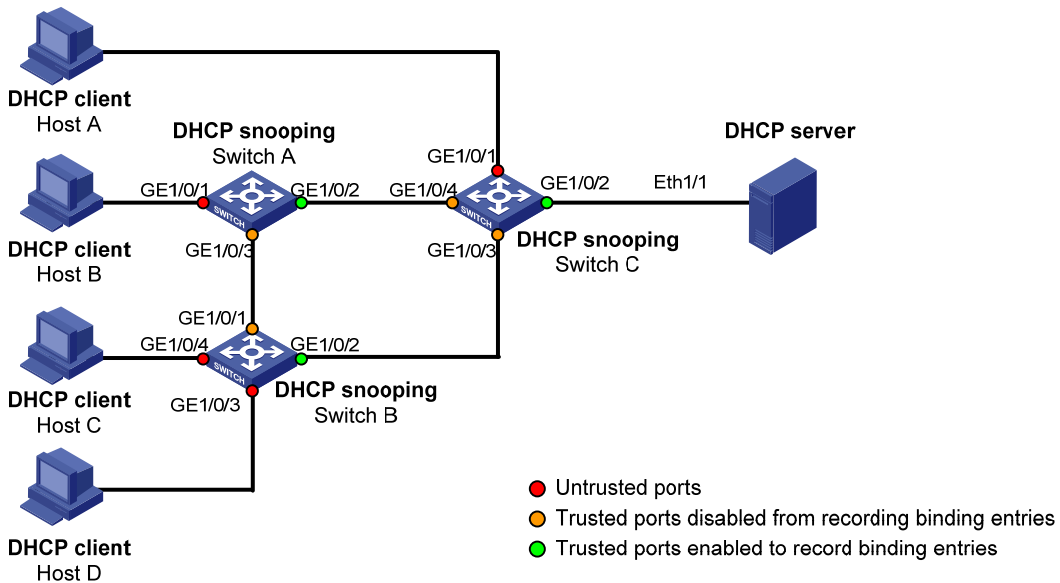


Configuring trusted ports in a cascaded network

In a cascaded network involving multiple DHCP snooping devices, the ports connected to other DHCP snooping devices should be configured as trusted ports.

To save system resources, you can disable the trusted ports, which are indirectly connected to DHCP clients, from recording client IP-to-MAC bindings upon receiving DHCP requests.

Figure 36 Configuring trusted ports in a cascaded network



DHCP snooping support for Option 82

Option 82 records the location information about the DHCP client so the administrator can locate the DHCP client for security control and accounting purposes. For more information, see "[Configuring DHCP relay agent.](#)"

If DHCP snooping supports Option 82, it handles a client's request according to the contents defined in Option 82, if any. The handling strategies are described in [Table 4](#).

If a reply returned by the DHCP server contains Option 82, the DHCP snooping device removes the Option 82 before forwarding the reply to the client. If the reply contains no Option 82, the DHCP snooping device forwards it directly.

Table 4 Handling strategies of DHCP snooping

If a client's requesting message has...	Handling strategy	Padding format	The DHCP snooping device...
Option 82	Drop	N/A	Drops the message.
	Keep	Random	Forwards the message without changing Option 82.
		normal	Forwards the message after replacing the original Option 82 with the Option 82 padded in normal format.
		verbose	Forwards the message after replacing the original Option 82 with the Option 82 padded in verbose format.
	Replace	user-defined	Forwards the message after replacing the original Option 82 with the user-defined Option 82.
		normal	Forwards the message without changing Option 82.
		verbose	Forwards the message without changing Option 82.
	Append	private	Forwards the message after adding sub-option 9 to option 82 or adding content to sub-option 9 that option 82 contains.
		standard	Forwards the message without changing Option 82.
		user-defined	Forwards the message without changing Option 82.
N/A		normal	Forwards the message after adding the Option 82 padded in normal format.

If a client's requesting message has...	Handling strategy	Padding format	The DHCP snooping device...
	N/A	private	Forwards the message after adding the Option 82 padded in private format.
	N/A	standard	Forwards the message after adding the Option 82 padded in standard format.
	N/A	verbose	Forwards the message after adding the Option 82 padded in verbose format.
	N/A	user-defined	Forwards the message after adding the user-defined Option 82.

The handling strategy and padding format for Option 82 on the DHCP snooping device are the same as those on the relay agent.

DHCP snooping configuration task list

Task	Remarks
Configuring DHCP snooping basic functions	Required
Configuring DHCP snooping to support Option 82	Optional
Configuring DHCP snooping entries backup	Optional
Enabling DHCP starvation attack protection	Optional
Enabling DHCP-REQUEST message attack protection	Optional
Configuring DHCP packet rate limit	Optional

Configuring DHCP snooping basic functions

Follow these guidelines when configure DHCP snooping basic functions:

- You must specify the ports connected to the authorized DHCP servers as trusted to make sure that DHCP clients can obtain valid IP addresses. The trusted port and the port connected to the DHCP client must be in the same VLAN.
- You can specify Layer 2 Ethernet interfaces and Layer 2 aggregate interfaces as trusted ports. For more information about aggregate interfaces, see *Layer 2—LAN Switching Configuration Guide*.
- If a Layer 2 Ethernet interface is added to an aggregation group, the DHCP snooping configuration of the interface will not take effect. After the interface quits the aggregation group, the configuration will be effective.
- DHCP snooping can work with basic QinQ or flexible QinQ. When receiving a packet without any VLAN tag from the DHCP client to the DHCP server, the DHCP snooping device adds a VLAN tag to the packet. If the packet has one VLAN tag, the device adds another VLAN tag to the packet and records the two VLAN tags in a DHCP snooping entry. The newly added VLAN tag is the outer tag. If the packet has two VLAN tags, the device directly forwards the packet to the DHCP server without adding any tag.
- If you need to add a new VLAN tag and meanwhile modify the original VLAN tag for the packet, DHCP snooping cannot work with flexible QinQ.

To configure DHCP snooping basic functions:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable DHCP snooping.	dhcp-snooping	Disabled by default.
3. Enter Ethernet interface view.	interface <i>interface-type</i> <i>interface-number</i>	The interface connects to the DHCP server.
4. Specify the port as a trusted port that records the IP-to-MAC bindings of clients.	dhcp-snooping trust	After DHCP snooping is enabled, a port is an untrusted port by default.
5. Return to system view.	quit	N/A
6. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	The interface indirectly connects to the DHCP client.
7. Specify the port as a trusted port that does not record the IP-to-MAC bindings of clients.	dhcp-snooping trust no-user-binding	Optional. After DHCP snooping is enabled, a port is an untrusted port by default.

Configuring DHCP snooping to support Option 82

Follow these guidelines when configure DHCP snooping to support Option 82:

- You can only enable DHCP snooping to support Option 82 on Layer 2 Ethernet interfaces, and Layer 2 aggregate interfaces.
- If a Layer 2 Ethernet interface is added to an aggregation group, enabling DHCP snooping to support Option 82 on the interface will not take effect. After the interface quits the aggregation group, the configuration will be effective.
- Option 82 support requires configuration on both the DHCP server and the device enabled with DHCP snooping. See "[Configuring DHCP server](#)" for DHCP server configuration of this kind.
- If the handling strategy of the DHCP-snooping-enabled device is configured as **replace**, you need to configure a padding format for Option 82. If the handling strategy is **keep** or **drop**, you need not configure any padding format.
- If the Option 82 is padded with the device name, the device name must contain no spaces. Otherwise, the DHCP-snooping device will drop the message. You can use the **sysname** command to specify the device name. For more information about this command, see *Fundamentals Command Reference*.
- If DHCP snooping and QinQ work together or the DHCP snooping device receives a DHCP packet with two VLAN tags, and the normal or verbose padding format is adopted for Option 82, DHCP snooping fills the VLAN ID field of sub-option 1 with outer VLAN tag.inter VLAN tag. For example, if the outer VLAN tag is 10 (a in hexadecimal) and the inner VLAN tag is 20 (14 in hexadecimal), the VLAN ID is 000a.0014.

To configure DHCP snooping to support Option 82:

Step	Command	Remarks
1. Enter system view.	system-view	N/A

Step	Command	Remarks
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Enable DHCP snooping to support Option 82.	dhcp-snooping information enable	Disabled by default.
4. Configure the handling strategy for requests containing Option 82.	dhcp-snooping information strategy { append drop keep replace }	Optional. replace by default.
5. Configure Option 82 in the non-user-defined padding format.	<ul style="list-style-type: none"> Configure the padding format for Option 82: dhcp-snooping information format { normal private <i>private</i> standard verbose [node-identifier { mac sysname user-defined <i>node-identifier</i> }] } Configure the code type for the circuit ID sub-option: dhcp-snooping information circuit-id format-type { ascii hex } Configure the code type for the remote ID sub-option: dhcp-snooping information remote-id format-type { ascii hex } Enable sub-option 9: dhcp-snooping information [vlan <i>vlan-id</i>] sub-option <i>sub-option-code</i> 	<p>Optional.</p> <p>By default:</p> <ul style="list-style-type: none"> The padding format for Option 82 is normal. The code type for the circuit ID sub-option depends on the padding format of Option 82. Each field has its own code type. The code type for the remote ID sub-option is hex. Sub-option 9 is not enabled <p>Hex configuration applies to private padding format only.</p> <p>The code type configuration for the circuit ID sub-option and remote ID sub-option apply to non-user-defined Option 82 only.</p> <p>For sub-option 9, when append strategy is adopted, the sysname and the primary IP address of the Loopback0 interface are padded. When some other strategy is adopted, only the sysname is padded.</p>
6. Configure user-defined Option 82.	<ul style="list-style-type: none"> Configure the padding content for the circuit ID sub-option: dhcp-snooping information [vlan <i>vlan-id</i>] circuit-id string <i>circuit-id</i> Configure the padding content for the remote ID sub-option: dhcp-snooping information [vlan <i>vlan-id</i>] remote-id string { <i>remote-id</i> sysname } Configure the padding content for the sub-option 9: dhcp-snooping information [vlan <i>vlan-id</i>] sub-option <i>sub-option-code</i> [string <i>user-string</i>&<1-8>] 	<p>Optional.</p> <p>By default:</p> <ul style="list-style-type: none"> The padding content for the circuit ID sub-option depends on the padding format of Option 82. The padding content for the remote ID sub-option depends on the padding format of Option 82. Sub-option 9 is not padded.

Configuring DHCP snooping entries backup

DHCP snooping entries cannot survive a reboot. If the DHCP snooping device is rebooted, security modules (such as IP source guard) that use DHCP snooping entries to authenticate users will reject requests from clients until new entries are learned.

The DHCP snooping entries backup feature enables you to store DHCP snooping entries in a file. When the DHCP snooping device reboots, it reads DHCP snooping entries from this file.

After DHCP snooping is disabled with the **undo dhcp-snooping** command, the device will delete all DHCP snooping entries, including those stored in the file.

To configure DHCP snooping entries backup:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Specify the name of the file for storing DHCP snooping entries.	dhcp-snooping binding database filename <i>filename</i>	Not specified by default. DHCP snooping entries are stored immediately after this command is used and then updated at the interval set by the dhcp-snooping binding database update interval command.
3. Back up DHCP snooping entries to the file.	dhcp-snooping binding database update now	Optional. DHCP snooping entries will be stored to the file each time this command is used.
4. Set the interval at which the DHCP snooping entry file is refreshed.	dhcp-snooping binding database update interval <i>minutes</i>	Optional. By default, the file is not refreshed periodically.

Enabling DHCP starvation attack protection

A DHCP starvation attack occurs when an attacker constantly sends forged DHCP requests using different MAC addresses in the **chaddr** field to a DHCP server. This exhausts the IP address resources of the DHCP server so legitimate DHCP clients cannot obtain IP addresses. The DHCP server may also fail to work because of exhaustion of system resources. You can protect against starvation attacks in the following ways:

- To relieve a DHCP starvation attack that uses DHCP packets encapsulated with different source MAC addresses, you can limit the number of MAC addresses that a Layer 2 port can learn.
- To prevent a DHCP starvation attack that uses DHCP requests encapsulated with the same source MAC address, enable MAC address check on the DHCP snooping device. With this function enabled, the DHCP snooping device compares the **chaddr** field of a received DHCP request with the source MAC address field of the frame. If they are the same, the request is considered valid and forwarded to the DHCP server. If not, the request is discarded.

Enable MAC address check only on Layer 2 Ethernet interfaces and Layer 2 aggregate interfaces.

To enable MAC address check:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Enable MAC address check.	dhcp-snooping check mac-address	Disabled by default

Enabling DHCP-REQUEST message attack protection

Attackers may forge DHCP-REQUEST messages to renew the IP address leases for legitimate DHCP clients that no longer need the IP addresses. These forged messages keep a victim DHCP server renewing the leases of IP addresses instead of releasing the IP addresses. This wastes IP address resources.

To prevent such attacks, you can enable DHCP-REQUEST message check on DHCP snooping devices. With this feature enabled, upon receiving a DHCP-REQUEST message, a DHCP snooping device looks up local DHCP snooping entries for the corresponding entry of the message. If an entry is found, the DHCP snooping device compares the entry with the message information. If they are consistent, the DHCP-REQUEST message is considered a valid lease renewal request and forwarded to the DHCP server. If they are not consistent, the message is considered a forged lease renewal request and discarded. If no corresponding entry is found, the message is considered valid and forwarded to the DHCP server.

Enable DHCP-REQUEST message check only on Layer 2 Ethernet interfaces, and Layer 2 aggregate interfaces.

To enable DHCP-REQUEST message check:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Enable DHCP-REQUEST message check.	dhcp-snooping check request-message	Disabled by default

Configuring DHCP packet rate limit

- You can configure DHCP packet rate limit only on Layer 2 Ethernet interfaces and Layer 2 aggregate interfaces.
- If a Layer 2 Ethernet interface belongs to an aggregation group, it uses the DHCP packet maximum rate configured on the corresponding Layer 2 aggregate interface.
- To identify DHCP packets from unauthorized DHCP servers, DHCP snooping delivers all incoming DHCP packets to the CPU. If a malicious user sends a large number of DHCP requests to the DHCP snooping device, the CPU of the device will be overloaded, and the device may even crash. To solve this problem, you can configure DHCP packet rate limit on relevant interfaces.

To configure DHCP packet rate limit:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter Layer 2 Ethernet interface view or Layer 2 aggregate interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Configure the maximum rate of incoming DHCP packets.	dhcp-snooping rate-limit <i>rate</i>	Not configured by default

Displaying and maintaining DHCP snooping

Task	Command	Remarks
Display DHCP snooping entries.	display dhcp-snooping [ip <i>ip-address</i>] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display Option 82 configuration information on the DHCP snooping device.	display dhcp-snooping information { all interface <i>interface-type</i> <i>interface-number</i> } [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display DHCP packet statistics on the DHCP snooping device .	display dhcp-snooping packet statistics [slot <i>slot-number</i>] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display information about trusted ports.	display dhcp-snooping trust [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display the DHCP snooping entry file information.	display dhcp-snooping binding database [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Clear DHCP snooping entries.	reset dhcp-snooping { all ip <i>ip-address</i> }	Available in user view
Clear DHCP packet statistics on the DHCP snooping device .	reset dhcp-snooping packet statistics [slot <i>slot-number</i>]	Available in user view

DHCP snooping configuration examples

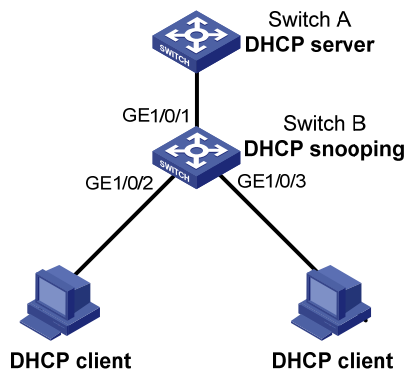
DHCP snooping configuration example

Network requirements

As shown in [Figure 37](#), Switch B is connected to a DHCP server through GigabitEthernet 1/0/1, and to two DHCP clients through GigabitEthernet 1/0/2 and GigabitEthernet 1/0/3. GigabitEthernet 1/0/1 forwards DHCP server responses while the other two do not.

Switch B records clients' IP-to-MAC address bindings in DHCP-REQUEST messages and DHCP-ACK messages received from trusted ports.

Figure 37 Network diagram



Configuration procedure

```
# Enable DHCP snooping.
<SwitchB> system-view
[SwitchB] dhcp-snooping

# Specify GigabitEthernet 1/0/1 as trusted.
[SwitchB] interface GigabitEthernet 1/0/1
[SwitchB-GigabitEthernet1/0/1] dhcp-snooping trust
[SwitchB-GigabitEthernet1/0/1] quit
```

DHCP snooping Option 82 support configuration example

Network requirements

As shown in [Figure 37](#), enable DHCP snooping and Option 82 support on Switch B.

- Configure the handling strategy for DHCP requests containing Option 82 as **replace**.
- On GigabitEthernet 1/0/2, configure the padding content for the circuit ID sub-option as **company001** and for the remote ID sub-option as **device001**.
- On GigabitEthernet 1/0/3, configure the padding format as **verbose**, access node identifier as **sysname**, and code type as **ascii** for Option 82.
- Switch B forwards DHCP requests to the DHCP server (Switch A) after replacing Option 82 in the requests, so that the DHCP clients can obtain IP addresses.

Configuration procedure

```
# Enable DHCP snooping.
<SwitchB> system-view
[SwitchB] dhcp-snooping

# Specify GigabitEthernet 1/0/1 as trusted.
[SwitchB] interface GigabitEthernet 1/0/1
[SwitchB-GigabitEthernet1/0/1] dhcp-snooping trust
[SwitchB-GigabitEthernet1/0/1] quit

# Configure GigabitEthernet 1/0/2 to support Option 82.
[SwitchB] interface GigabitEthernet 1/0/2
[SwitchB-GigabitEthernet1/0/2] dhcp-snooping information enable
```

```
[SwitchB-GigabitEthernet1/0/2] dhcp-snooping information strategy replace
[SwitchB-GigabitEthernet1/0/2] dhcp-snooping information circuit-id string company001
[SwitchB-GigabitEthernet1/0/2] dhcp-snooping information remote-id string device001
[SwitchB-GigabitEthernet1/0/2] quit

# Configure GigabitEthernet 1/0/3 to support Option 82.
[SwitchB] interface GigabitEthernet 1/0/3
[SwitchB-GigabitEthernet1/0/3] dhcp-snooping information enable
[SwitchB-GigabitEthernet1/0/3] dhcp-snooping information strategy replace
[SwitchB-GigabitEthernet1/0/3] dhcp-snooping information format verbose node-identifier
sysname
[SwitchB-GigabitEthernet1/0/3] dhcp-snooping information circuit-id format-type ascii
[SwitchB-GigabitEthernet1/0/3] dhcp-snooping information remote-id format-type ascii
```

Configuring BOOTP client

Overview

BOOTP application

After you specify an interface of a device as a BOOTP client, the interface can use BOOTP to get information (such as IP address) from the BOOTP server.

To use BOOTP, an administrator must configure a BOOTP parameter file for each BOOTP client on the BOOTP server. The parameter file contains information such as MAC address and IP address of a BOOTP client. When a BOOTP client sends a request to the BOOTP server, the BOOTP server searches for the BOOTP parameter file and returns the corresponding configuration information.

BOOTP is usually used in relatively stable environments. In network environments that change frequently, DHCP is more suitable.

Because a DHCP server can interact with a BOOTP client, you can use the DHCP server to configure an IP address for the BOOTP client, without any BOOTP server.

Obtaining an IP address dynamically

A BOOTP client dynamically obtains an IP address from a BOOTP server in the following steps:

1. The BOOTP client broadcasts a BOOTP request, which contains its own MAC address.
2. The BOOTP server receives the request and searches the configuration file for the corresponding IP address and other information according to the MAC address of the BOOTP client. The BOOTP server then returns a BOOTP response to the BOOTP client.
3. The BOOTP client obtains the IP address from the received response.

A DHCP server can take the place of the BOOTP server in the above mentioned dynamic IP address acquisition.

Protocols and standards

- RFC 951, *Bootstrap Protocol (BOOTP)*
- RFC 2132, *DHCP Options and BOOTP Vendor Extensions*
- RFC 1542, *Clarifications and Extensions for the Bootstrap Protocol*

Configuration restrictions

- BOOTP client configuration only applies to VLAN interfaces.
- If several VLAN interfaces sharing the same MAC address obtain IP addresses through a BOOTP relay agent, the BOOTP server cannot be a Windows Server 2000 or Windows Server 2003.

Configuring an interface to dynamically obtain an IP address through BOOTP

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Configure an interface to dynamically obtain an IP address through BOOTP.	ip address bootp-alloc	By default, an interface does not use BOOTP to obtain an IP address.

Displaying and maintaining BOOTP client configuration

Task	Command	Remarks
Display BOOTP client information.	display bootp client [interface <i>interface-type</i> <i>interface-number</i>] [{ begin exclude include } <i>regular-expression</i>]	Available in any view

BOOTP client configuration example

Network requirements

As shown in [Figure 29](#), Switch B's port belonging to VLAN 1 connects to the LAN. VLAN-interface 1 obtains an IP address from the DHCP server by using BOOTP.

Configuration procedure

The following describes only the configuration on Switch B serving as a client.

Configure VLAN-interface 1 to dynamically obtain an IP address from the DHCP server.

```
<SwitchB> system-view
[SwitchB] interface vlan-interface 1
[SwitchB-Vlan-interface1] ip address bootp-alloc
```

Use the **display bootp client** command to view the IP address assigned to the BOOTP client.

To make the BOOTP client obtain an IP address from the DHCP server, you must perform additional configurations on the DHCP server. For more information, see "[Configuring DHCP server.](#)"

Configuring IPv4 DNS

Overview

Domain Name System (DNS) is a distributed database used by TCP/IP applications to translate domain names into corresponding IP addresses. With DNS, you can use easy-to-remember domain names in some applications and let the DNS server translate them into correct IP addresses.

DNS services can be static or dynamic. After a user specifies a name, the device checks the local 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 setting up mappings between domain names and IP addresses. IP addresses of the corresponding domain names can be found in the static domain resolution table when you use applications such as Telnet.

Dynamic domain name resolution

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 a higher level DNS server. 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 application.

Figure 38 Dynamic domain name resolution

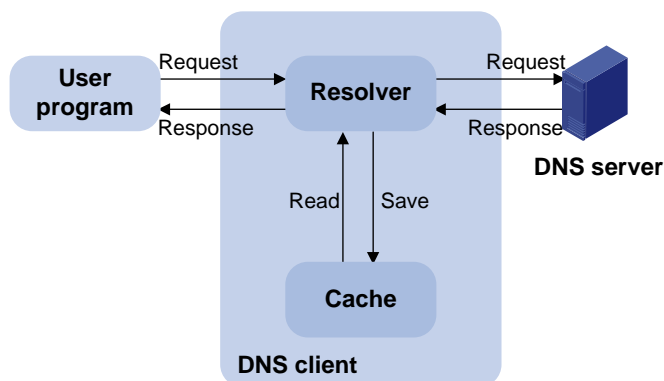


Figure 38 shows the relationship between the user program, DNS client, and DNS server.

The DNS client is made up of the resolver and cache. The user program and DNS client can run on the same device or different devices, but the DNS server and the DNS client usually run on different devices.

Dynamic domain name resolution allows the DNS client to store latest mappings between domain names and IP addresses in the dynamic domain name cache. The DNS client does not need to send a request to the DNS server for a repeated query next time. The aged mappings are removed from the cache after some time, and latest entries are required from the DNS server. The DNS server decides how long a mapping is valid, and the DNS client gets the aging information from DNS messages.

DNS suffixes

The DNS client holds a list of suffixes which the user sets. The resolver can use the list to supply the missing part of incomplete names.

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

- If there is no dot (.) in the domain name (for example, aabbcc), the resolver considers this a host name and adds a DNS suffix before the query. If no match is found after all the configured suffixes are used, the original domain name (for example, aabbcc) is used for the query.
- If there is a dot (.) in the domain name (for example, www.aabbcc), the resolver directly uses this domain name for the query. If the query fails, the resolver adds a DNS suffix for another query.
- If the dot (.) is at the end of the domain name (for example, aabbcc.com.), the resolver considers it a Fully Qualified Domain Name (FQDN) and returns the query result, successful or failed. The dot (.) is considered a terminating symbol.

The device supports static and dynamic DNS client services.

NOTE:

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.

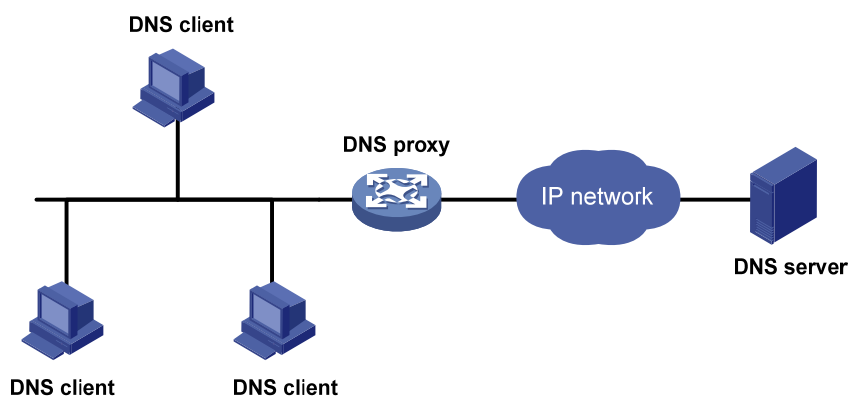
DNS proxy

A DNS proxy forwards DNS requests and replies between DNS clients and a DNS server.

As shown in [Figure 39](#), a DNS client sends a DNS request to the DNS proxy, which forwards the request to the designated DNS server, and 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 on only the DNS proxy instead of on each DNS client.

Figure 39 DNS proxy networking 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 table 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.

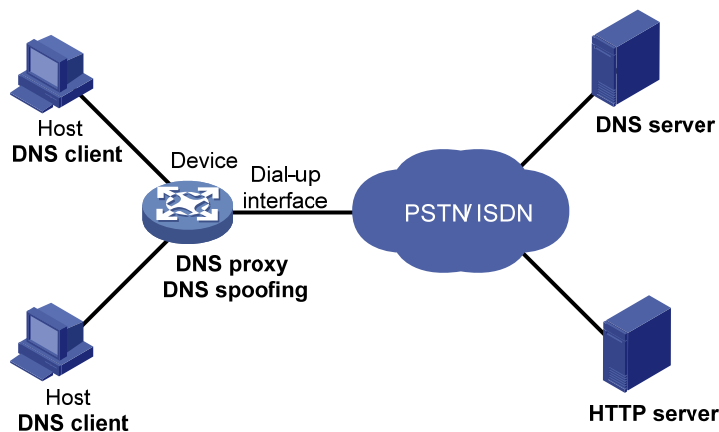
With no DNS server or route to a DNS server specified, the DNS proxy does not forward DNS requests, or answer requests from the DNS clients.

DNS spoofing

DNS spoofing is applied to the dial-up network, as shown in [Figure 40](#).

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

Figure 40 Application of DNS spoofing



Without DNS spoofing enabled, the device forwards the DNS requests received from the hosts to the DNS server, if it cannot find a match in the local domain name resolution table. However, without any dial-up connection established, the device cannot obtain the DNS server address, so it cannot forward or answer the requests from the clients. The domain name cannot be resolved and no traffic triggers the establishment of a dial-up connection.

DNS spoofing can solve this problem. DNS spoofing enables the device to reply the DNS client with a configured IP address when the device does not have a DNS server address or route to a DNS server. Subsequent packets sent by the DNS client trigger the establishment of a dial-up connection with the network.

In the network of [Figure 40](#), a host accesses the HTTP server in following these 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. If no match is found and the device does know the DNS server address, the device spoofs the host by replying a configured IP address. The TTL of the DNS reply is 0. The device must have a route to the IP address with the dial-up interface as the outgoing interface.
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 establishes a dial-up connection with the network and dynamically obtains the DNS server address through DHCP or other autoconfiguration mechanisms.
5. When the DNS reply ages out, the host sends a DNS request to the device again.
6. Then the device operates the same as a DNS proxy. For more information, see "[A DNS proxy operates as follows:](#)"
7. After obtaining the IP address of the HTTP server, the host can access the HTTP server.

Because the IP address configured with DNS spoofing is not the actual IP address of the requested domain name, the TTL of the DNS reply is set to 0 to prevent the DNS client from generating incorrect domain name-to-IP address mappings.

Configuring the IPv4 DNS client

Configuring static domain name resolution

Configuring static domain name resolution refers to specifying the mappings between host names and IPv4 addresses. Static domain name resolution allows applications such as Telnet to contact hosts by using host names instead of IPv4 addresses.

Follow these guidelines when you configure static domain name resolution:

- The IPv4 address you last assign to the host name will overwrite the previous one if there is any.
- You may create up to 50 static mappings between domain names and IPv4 addresses.

To configure static domain name resolution:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Configure a mapping between a host name and an IPv4 address.	ip host <i>hostname ip-address</i>	Not configured by default

Configuring dynamic domain name resolution

To send DNS queries to a correct server for resolution, dynamic domain name resolution needs to be enabled and a DNS server needs to be configured.

In addition, you can configure a DNS suffix that the system will automatically add to the provided domain name for resolution.

Configuration restrictions and guidelines

- You can configure up to six DNS servers, including those with IPv6 addresses, in system view, and up to six DNS servers on all interfaces of a device.
- A DNS server configured in system view has a higher priority than one configured in interface view. A DNS server configured earlier has a higher priority than one configured later in the same view. A DNS server manually configured has a higher priority than one dynamically obtained through DHCP. A name query request is first sent to the DNS server that has the highest priority. If no reply is received, it is sent to the DNS server that has the second highest priority, and thus in turn.
- You can specify up to ten DNS suffixes.

Configuration procedure

To configure dynamic domain name resolution:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable dynamic domain name resolution.	dns resolve	Disabled by default.
3. Specify a DNS server.	<ul style="list-style-type: none">• (Approach 1) In system view: dns server ip-address• (Approach 2) In interface view:<ul style="list-style-type: none">a. interface interface-type interface-numberb. dns server ip-addressc. quit	Use at least one approach. Not specified by default.
4. Configure a DNS suffix.	dns domain domain-name	Optional. Not configured by default. Only the provided domain name is resolved.

Configuring the DNS proxy

You can specify multiple DNS servers by using the **dns server** command repeatedly. Upon receiving a name query request from a client, the DNS proxy forwards the 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 thus in turn.

To configure the DNS proxy:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable DNS proxy.	dns proxy enable	Disabled by default.

Step	Command	Remarks
3. Specify a DNS server.	<ul style="list-style-type: none"> (Method 1) In system view: dns server <i>ip-address</i> (Method 2) In interface view: <ul style="list-style-type: none"> a. interface <i>interface-type</i> <i>interface-number</i> b. dns server <i>ip-address</i> 	<p>Use at least one method.</p> <p>No DNS server is specified by default.</p>

Configuring DNS spoofing

DNS spoofing is effective only when:

- The DNS proxy is enabled on the device.
- No DNS server or route to any DNS server is specified on the device.

To configure DNS spoofing:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable DNS spoofing and specify the translated IP address.	dns spoofing <i>ip-address</i>	Disabled by default

Setting the DSCP value for DNS packets

An IPv4 packet header contains an 8-bit Type of Service (ToS) field. As defined in RFC 2474, the first six bits set the Differentiated Services Code Point (DSCP) value, and the last two bits are reserved. Network devices use the DSCP value as a reference to determine the packet priority for transmission.

To set the DSCP value for DNS packets:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Set the DSCP value for DNS packets.	dns dscp <i>dscp-value</i>	<p>Optional.</p> <p>By default, the DSCP value for DNS packets is 0.</p>

Specifying the source interface for DNS packets

By default, the device uses the primary IP address of the output interface of the matching route as the source IP address of a DNS request. Therefore, the source IP address of the DNS packets may vary with DNS servers. In some scenarios, the DNS server only responds to DNS requests sourced from a specific IP address. In such cases, you must specify the source interface for the DNS packets so that the device can always use the primary IP address of the specified source interface as the source IP address of DNS packets.

To specify the source interface for DNS packets:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Set the DSCP value for DNS packets.	dns source-interface <i>interface-type interface-number</i>	By default, no source interface for DNS packets is specified. The device uses the primary IP address of the output interface of the matching route as the source IP address of a DNS request.

Displaying and maintaining IPv4 DNS

Task	Command	Remarks
Display the static IPv4 domain name resolution table.	display ip host [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display IPv4 DNS server information.	display dns server [dynamic] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display DNS suffixes.	display dns domain [dynamic] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display information about the dynamic IPv4 domain name cache.	display dns host ip [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Clear information about the dynamic IPv4 domain name cache.	reset dns host ip	Available in user view

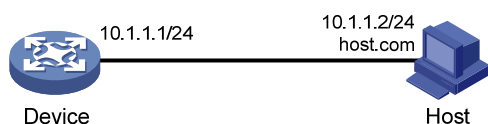
Static domain name resolution configuration example

Network requirements

As shown in [Figure 41](#), the device wants to access the host by using an easy-to-remember domain name rather than an IP address.

Configure static domain name resolution on the device so that the device can use the domain name `host.com` to access the host whose IP address is `10.1.1.2`.

Figure 41 Network diagram



Configuration 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
```

Use the **ping host.com** command to 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
  Reply from 10.1.1.2: bytes=56 Sequence=1 ttl=128 time=1 ms
  Reply from 10.1.1.2: bytes=56 Sequence=2 ttl=128 time=4 ms
  Reply from 10.1.1.2: bytes=56 Sequence=3 ttl=128 time=3 ms
  Reply from 10.1.1.2: bytes=56 Sequence=4 ttl=128 time=2 ms
  Reply from 10.1.1.2: bytes=56 Sequence=5 ttl=128 time=3 ms

--- host.com ping statistics ---
 5 packet(s) transmitted
 5 packet(s) received
 0.00% packet loss
 round-trip min/avg/max = 1/2/4 ms
```

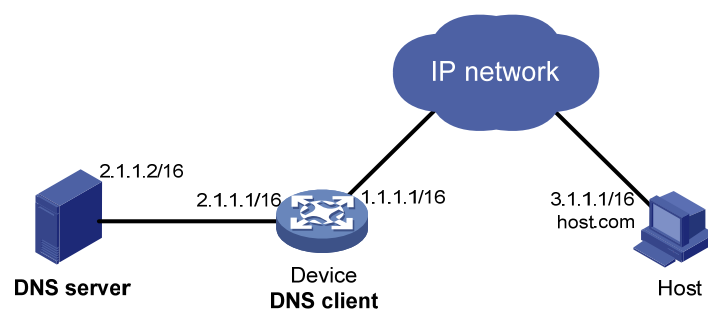
Dynamic domain name resolution configuration example

Network requirements

As shown in [Figure 42](#), the device wants to access the host by using an easy-to-remember domain name rather than an IP address, and to request the DNS server on the network for an IP address by using dynamic domain name resolution. The IP address of the DNS server is 2.1.1.2/16 and the DNS server has a com domain, which stores the mapping between domain name host and IP address 3.1.1.1/16.

Configure dynamic domain name resolution and the domain name suffix com on the device that serves as a DNS client so that the device can use domain name host to access the host with the domain name host.com and the IP address 3.1.1.1/16.

Figure 42 Network diagram



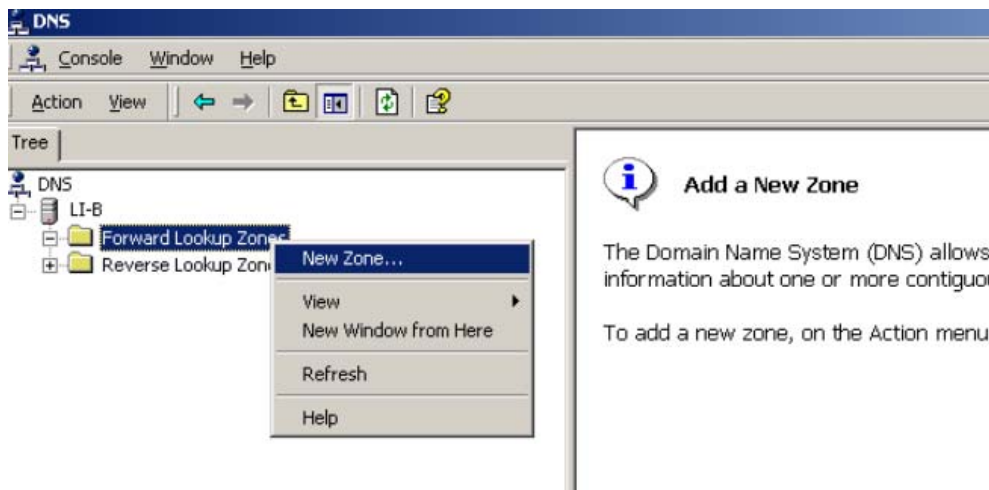
Configuration procedure

Before performing the following configuration, make sure the device and the host are accessible to each other via available routes, and that the IP addresses of the interfaces are configured as shown [Figure 42](#).

This configuration may vary with DNS servers. The following configuration is performed on a PC running Windows Server 2000.

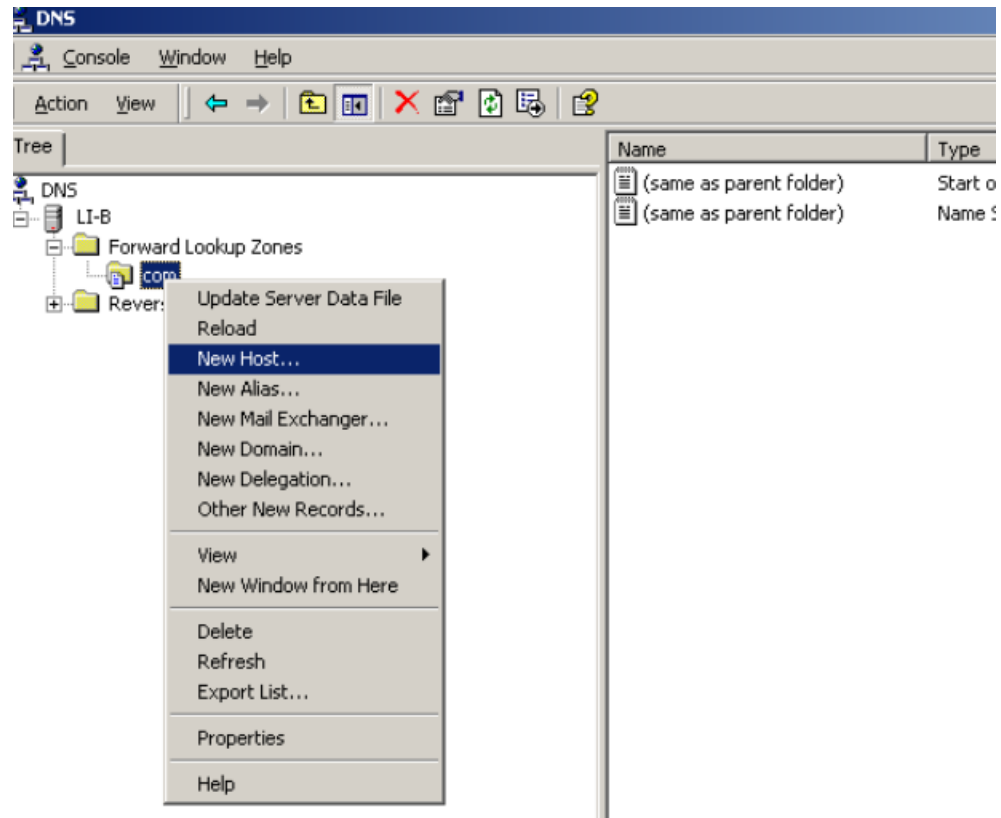
1. Configure the DNS server:
 - a. Select **Start > Programs > Administrative Tools > DNS**.
The DNS server configuration page appears, as shown in [Figure 43](#).
 - b. Right click **Forward Lookup Zones**, select **New Zone**, and then follow the steps to create a new zone named **com**.

Figure 43 Creating a zone



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

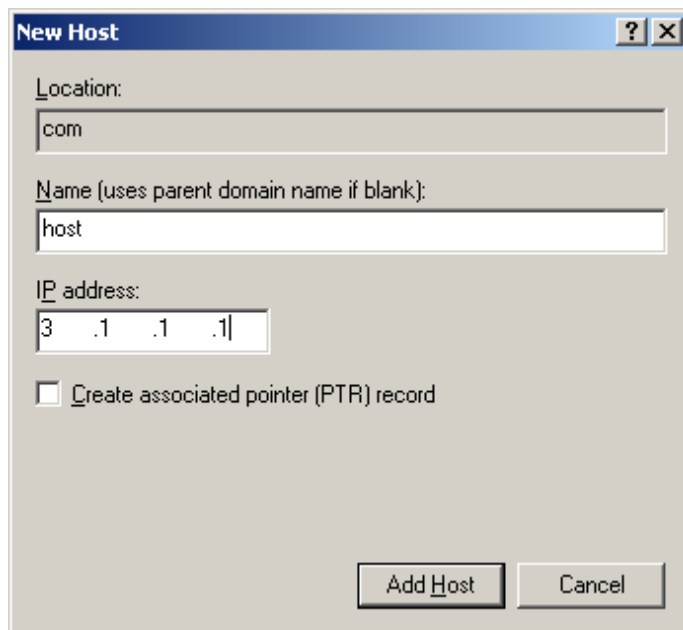
Figure 44 Adding a host



- d. On the page that appears, enter host name **host** and IP address **3.1.1.1**.
- e. Click **Add Host**.

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

Figure 45 Adding a mapping between domain name and IP address



- 2. Configure the DNS client:


```

# Enable dynamic domain name resolution.
<Sysname> system-view
[Sysname] dns resolve
# Specify the DNS server 2.1.1.2.
[Sysname] dns server 2.1.1.2
# Configure com as the name suffix.
[Sysname] dns domain com

```

Verifying the configuration

Use the **ping host** command on the device to verify that the communication between the device and the host is normal and that the corresponding destination IP address is 3.1.1.1.

```

[Sysname] ping host
Trying DNS resolve, press CTRL_C to break
Trying DNS server (2.1.1.2)
PING host.com (3.1.1.1):
56 data bytes, press CTRL_C to break
  Reply from 3.1.1.1: bytes=56 Sequence=1 ttl=126 time=3 ms
  Reply from 3.1.1.1: bytes=56 Sequence=2 ttl=126 time=1 ms
  Reply from 3.1.1.1: bytes=56 Sequence=3 ttl=126 time=1 ms
  Reply from 3.1.1.1: bytes=56 Sequence=4 ttl=126 time=1 ms
  Reply from 3.1.1.1: bytes=56 Sequence=5 ttl=126 time=1 ms

--- host.com ping statistics ---
 5 packet(s) transmitted
 5 packet(s) received
 0.00% packet loss
 round-trip min/avg/max = 1/1/3 ms

```

DNS proxy configuration example

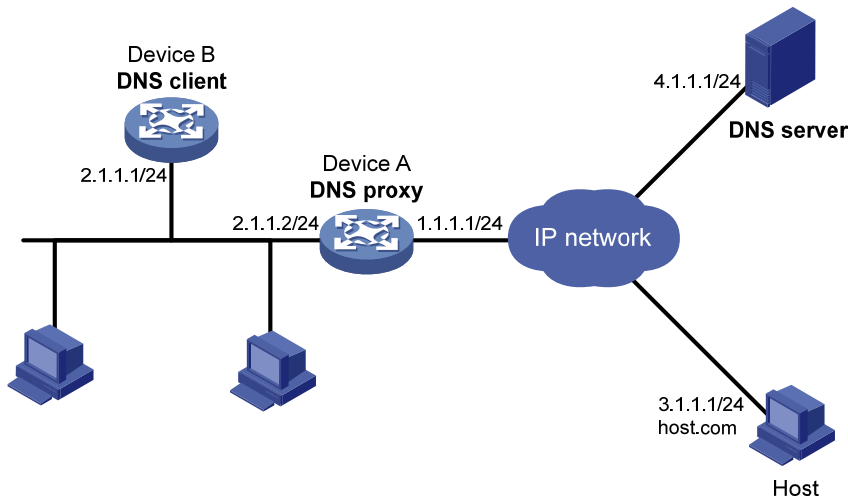
Network requirements

When the IP address of the DNS server changes, you must configure the new IP address of the DNS server on each device on the LAN. To simplify network management, you can use the DNS proxy function.

As shown in [Figure 46](#):

- Specify Device A as the DNS server of Device B (the DNS client). Device A acts as a DNS proxy. The IP address of the real DNS server is 4.1.1.1.
- Configure the IP address of the DNS proxy on Device B. DNS requests of Device B are forwarded to the real DNS server through the DNS proxy.

Figure 46 Network diagram



Configuration procedure

Before performing the following configuration, make sure Device A, the DNS server, and the host are reachable to each other and the IP addresses of the interfaces are configured as shown in [Figure 46](#).

1. Configure the DNS server:
This configuration may vary with different DNS servers. When a PC running Windows Server 2000 acts as the DNS server, see "[Dynamic domain name resolution configuration example](#)" for related 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:
Enable the domain name resolution function.
<DeviceB> system-view
[DeviceB] dns resolve
Specify the DNS server 2.1.1.2.
[DeviceB] dns server 2.1.1.2

Verifying the configuration

Execute the **ping host.com** command on Device B to verify that the communication between the device and the host is normal and that the corresponding destination IP address is 3.1.1.1.

```
[DeviceB] ping host.com
Trying DNS resolve, press CTRL_C to break
Trying DNS server (2.1.1.2)
PING host.com (3.1.1.1):
56 data bytes, press CTRL_C to break
```

```
Reply from 3.1.1.1: bytes=56 Sequence=1 ttl=126 time=3 ms
Reply from 3.1.1.1: bytes=56 Sequence=2 ttl=126 time=1 ms
Reply from 3.1.1.1: bytes=56 Sequence=3 ttl=126 time=1 ms
Reply from 3.1.1.1: bytes=56 Sequence=4 ttl=126 time=1 ms
Reply from 3.1.1.1: bytes=56 Sequence=5 ttl=126 time=1 ms

--- host.com ping statistics ---
 5 packet(s) transmitted
 5 packet(s) received
 0.00% packet loss
round-trip min/avg/max = 1/1/3 ms
```

Troubleshooting IPv4 DNS configuration

Symptom

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

Solution

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, verify 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 IP address is incorrect, verify 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.

Configuring IRDP

Overview

As an extension of the Internet Control Message Protocol (ICMP), the ICMP Router Discovery Protocol (IRDP) enables hosts to discover the IP addresses of their neighboring routers and set their default routes.

NOTE:

The hosts in this chapter support IRDP.

Background

Before a host can send packets to another network, it must know the IP address of at least one router on the local subnet. The host can obtain this information either through manual configuration, or from routing protocol packets sent by routers on the local subnet.

Both methods have disadvantages. The first method requires the administrator to manually configure and maintain router address information on hosts, and cannot track dynamic changes. The second method requires hosts to recognize various routing protocols, and will fail to work if no routing protocol runs on the local subnet.

IRDP was introduced to solve the problem. IRDP uses two new types of ICMP messages to allow hosts to discover neighboring routers. IRDP adapts to dynamic changes, requires less manual configuration, and does not rely on any routing protocols.

Working mechanism

IRDP uses the following types of ICMP messages.

- **Router advertisement (RA)**—Sent by a router to advertise its IP address and preference.
- **Router solicitation (RS)**—Sent by a host to voluntarily request the IP addresses of routers on the subnet.

IRDP operates in the following steps:

1. A router periodically broadcasts or multicasts an RA, which contains the IP addresses (including the primary IP address and manually configured secondary IP addresses) of interfaces. Hosts listen for RAs to obtain the IP addresses of neighboring routers.
2. Rather than wait for RAs, a newly attached host can voluntarily send an RS to request immediate RAs for the IP addresses of routers on the subnet. If no response to the RS is received, the host retransmits the RS several times. If the host still receives no RAs, it will obtain the IP addresses of routers from periodic RAs.
3. Upon receiving an RA, a host adds the IP addresses in the RA to its routing table. The host selects the IP address with the highest preference among all obtained IP addresses as the default gateway.

IRDP allows hosts to locate routers, but does not suggest the best route to a specific destination. If a host selects a router that is not the best next hop to a specific destination, the router will send back an ICMP redirect message to provide a better next hop.

Concepts

Preference of an IP address

Every IP address advertised in RAs has a preference value. The IP address with the highest preference is selected as the default router address.

You can configure the preference for IP addresses advertised on a router interface.

The bigger the preference value, the higher the preference. The minimum preference value (-2147483648) is used to indicate that the address, even though it may be advertised, is not to be used by neighboring hosts as a default router address.

Lifetime of an IP address

An RA contains a lifetime field that specifies the lifetime of advertised IP addresses. If no new RA for an IP address is received within the lifetime of the IP address, the host removes the corresponding route information.

All the IP addresses advertised by an interface have the same lifetime.

Advertising interval

A router interface with IRDP enabled sends out RAs at a random interval between the minimum advertising interval and the maximum advertising interval. This mechanism prevents the local link from being overloaded by a large number of RAs sent simultaneously from routers.

H3C recommends shortening the advertising interval on a link that suffers high packet loss rates.

Destination address of RAs

An RA uses either of the two destination IP addresses:

- broadcast address 255.255.255.255.
- Multicast address 224.0.0.1, which identifies all the hosts on the local subnet.

By default, the destination IP address of an RA is the broadcast address. If the interface that sends RAs supports multicast, configure 224.0.0.1 as the destination IP address.

Proxy-advertised IP addresses

By default, an interface advertises its primary IP address and manually configured secondary IP addresses. You can configure other IP addresses for an interface to proxy-advertise.

Protocols and standards

RFC 1256, *ICMP Router Discovery Messages*

Configuration procedure

IRDP configuration takes effect only when IRDP is enabled.

To configure IRDP:

Step	Command	Remarks
1. Enter system view.	system-view	N/A

Step	Command	Remarks
2. Enter VLAN interface view.	interface <i>vlan-interface</i> <i>vlan-interface-id</i>	N/A
3. Enable IRDP on the interface.	ip irdp	Disabled by default.
4. Configure the preference of advertised IP addresses.	ip irdp preference <i>preference-value</i>	Optional. The preference defaults to 0. The specified preference applies to all advertised IP addresses, including the primary IP address and the manually configured secondary IP addresses of the interface.
5. Set the lifetime of advertised IP addresses.	ip irdp lifetime <i>life-number</i>	Optional. 1800 seconds by default. The specified lifetime applies to all advertised IP addresses, including the IP address of the interface and proxy-advertised IP addresses on the interface.
6. Set the minimum advertising interval.	ip irdp minadvinterval <i>min-value</i>	Optional. 450 seconds by default.
7. Set the maximum advertising interval.	ip irdp maxadvinterval <i>max-value</i>	Optional. 600 seconds by default.
8. Configure the multicast address (224.0.0.1) as the destination IP address of RAs.	ip irdp multicast	Optional. By default, RAs use the broadcast address 255.255.255.255 as the destination IP address.
9. Specify a proxy-advertised IP address and its preference.	ip irdp address <i>ip-address</i> <i>preference</i>	Optional.

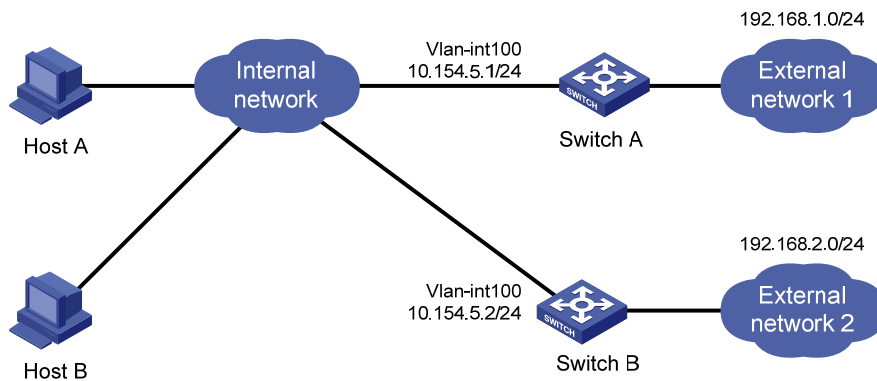
IRDP configuration example

Network requirements

Host A and Host B that run Linux operating systems reside in the internal network of a company. Switch A and Switch B serve as the egress routers and connect to external networks 192.168.1.0/24 and 192.168.2.0/24 respectively.

Configure Switch A as the default gateway of the hosts. The packets to the external networks can be properly routed.

Figure 47 Network diagram



Configuration procedure

1. Configure Switch A:

Specify the IP address for VLAN-interface 100.

```
<SwitchA> system-view
[SwitchA] interface Vlan-interface 100
[SwitchA-Vlan-interface100] ip address 10.154.5.1 24
```

Enable IRDP on VLAN-interface 100.

```
[SwitchA-Vlan-interface100] ip irdp
```

Specify preference 1000 for the IP address of VLAN-interface 100.

```
[SwitchA-Vlan-interface100] ip irdp preference 1000
```

Configure the multicast address 224.0.0.1 as the destination IP address for RAs sent by VLAN-interface 100.

```
[SwitchA-Vlan-interface100] ip irdp multicast
```

Specify the IP address 192.168.1.0 and preference 400 for VLAN-interface 100 to proxy-advertise.

```
[SwitchA-Vlan-interface100] ip irdp address 192.168.1.0 400
```

2. Configure Switch B:

Specify the IP address of VLAN-interface 100.

```
<SwitchB> system-view
[SwitchB] interface Vlan-interface 100
[SwitchB-Vlan-interface100] ip address 10.154.5.2 24
```

Enable IRDP on VLAN-interface 100.

```
[SwitchB-Vlan-interface100] ip irdp
```

Specify preference 500 for the IP address of VLAN-interface 100.

```
[SwitchB-Vlan-interface100] ip irdp preference 500
```

Configure the multicast address 224.0.0.1 as the destination IP address for RAs sent by VLAN-interface 100.

```
[SwitchB-Vlan-interface100] ip irdp multicast
```

Specify the IP address 192.168.2.0 and preference 400 for VLAN-interface 100 to proxy-advertise.

```
[SwitchB-Vlan-interface100] ip irdp address 192.168.2.0 400
```

Verifying the configuration

After enabling IRDP on Host A and Host B, display the routing table for the hosts (Host A for example).

```
[HostA@localhost ~]$ netstat -rne
```

```
Kernel IP routing table
```

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
10.154.5.0	0.0.0.0	255.255.255.0	U	0	0	0	eth1
192.168.1.0	0.0.0.0	255.255.255.0	U	0	0	0	eth1
192.168.2.0	0.0.0.0	255.255.255.0	U	0	0	0	eth1
0.0.0.0	10.154.5.1	0.0.0.0	UG	0	0	0	eth1

The output shows that the default route on Host A points to IP address 10.154.5.1, and Host A has routes to 192.168.1.0/24 and 192.168.2.0/24.

Optimizing IP performance

Enabling receiving and forwarding of directed broadcasts to a directly connected network

Directed broadcast packets are broadcast on a specific network. In the destination IP address of a directed broadcast, the network ID identifies the target network, and the host ID is made up of all ones. If a device is allowed to forward directed broadcasts to a directly connected network, hackers may mount attacks to the network. However, you can enable the feature by using the UDP Helper function to convert broadcasts to unicasts and forward them to a specified server.

Enabling receiving of directed broadcasts to a directly connected network

If the switch is enabled to receive directed broadcasts, the switch determines whether to forward them according to the configuration on the outgoing interface.

To enable the device to receive directed broadcasts:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable the device to receive directed broadcasts.	ip forward-broadcast	Disabled by default

Enabling forwarding of directed broadcasts to a directly connected network

Follow these guidelines when you enable the device to forward directed broadcasts:

- If an ACL is referenced in the **ip forward-broadcast** command, only packets permitted by the ACL can be forwarded.
- If you repeatedly execute the **ip forward-broadcast** command on an interface, only the last command takes effect. If the command executed last does not include **acl acl-number**, the ACL configured previously is removed.

To enable the device to forward directed broadcasts:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type interface-number</i>	N/A
3. Enable the interface to forward directed broadcasts.	ip forward-broadcast [acl <i>acl-number</i>]	Disabled by default

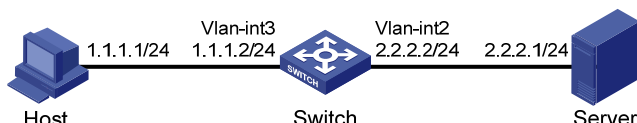
Configuration example

Network requirements

As shown in Figure 48, the host's interface and VLAN-interface 3 of the switch are on the same network segment (1.1.1.0/24). VLAN-interface 2 of Switch and the server are on another network segment (2.2.2.0/24). The default gateway of the host is VLAN-interface 3 (IP address 1.1.1.2/24) of Switch.

Configure the switch so that the server can receive directed broadcasts from the host to IP address 2.2.2.255.

Figure 48 Network diagram



Configuration procedure

Enable the switch to receive directed broadcasts.

```
<Switch> system-view  
[Switch] ip forward-broadcast
```

Configure IP addresses for VLAN-interface 3 and VLAN-interface 2.

```
[Switch] interface vlan-interface 3  
[Switch-Vlan-interface3] ip address 1.1.1.2 24  
[Switch-Vlan-interface3] quit  
[Switch] interface vlan-interface 2  
[Switch-Vlan-interface2] ip address 2.2.2.2 24
```

Enable VLAN-interface 2 to forward directed broadcasts.

```
[Switch-Vlan-interface2] ip forward-broadcast
```

Configuring TCP attributes

Configuring TCP path MTU discovery

! IMPORTANT:

All the devices on the TCP path must be enabled to send ICMP error messages by using the **ip unreachable enable** command.

TCP path MTU discovery (in RFC 1191) discovers the path MTU between the source and destination ends of a TCP connection. It works as follows:

1. A TCP source device sends a packet with the Don't Fragment (DF) bit set.
2. A router that fails to forward the packet because it exceeds the MTU on the outgoing interface discards the packet and returns an ICMP error message, which contains the MTU of the outgoing interface.
3. Upon receiving the ICMP message, the TCP source device calculates the current path MTU of the TCP connection.

4. The TCP source device sends subsequent TCP segments that each are smaller than the MSS (MSS = path MTU - IP header length - TCP header length).

If the TCP source device still receives ICMP error messages when the MSS is smaller than 32 bytes, the TCP source device will fragment packets.

An ICMP error message received from a router that does not support RFC 1191 has the MTU of the outgoing interface set to 0. Upon receiving the ICMP message, the TCP source device selects the path MTU smaller than the current path MTU from the MTU table as described in RFC 1191 to calculate the TCP MSS. The MTU table contains MTUs of 68, 296, 508, 1006, 1280, 1492, 2002, 4352, 8166, 17914, 32000, and 65535 bytes. Because the minimum TCP MSS specified by the system is 32 bytes, the actual minimum MTU is 72 bytes.

After you enable TCP path MTU discovery, all new TCP connections will detect the path MTU. The device uses the path MTU to calculate the MSS to avoid IP fragmentation.

The path MTU uses an aging mechanism to make sure that the source device can increase the path MTU when the minimum link MTU on the path increases.

- When the TCP source device receives an ICMP error message, it reduces the path MTU and starts an age timer for the path MTU.
- After the age timer expires, the source device uses a larger MSS in the MTU table as described in RFC 1191.
- If no ICMP error message is received within two minutes, the source device increases the MSS again until the MSS is as large as the MSS negotiated during TCP three-way handshake.

To enable TCP path MTU discovery:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable TCP path MTU discovery.	tcp path-mtu-discovery [aging <i>minutes</i> no-aging]	Optional. Disabled by default.

Configuring the TCP send/receive buffer size

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Configure the size of TCP send/receive buffer.	tcp window <i>window-size</i>	Optional. 8 KB by default.

Configuring TCP timers

You can configure the following TCP timers:

- **synwait timer**—When sending a SYN packet, TCP starts the synwait timer. If no response packet is received within the synwait timer interval, the TCP connection cannot be created.
- **finwait timer**—When a TCP connection is changed into FIN_WAIT_2 state, the finwait timer is started. If no FIN packet is received within the timer interval, the TCP connection is terminated. If a FIN packet is received, the TCP connection state changes to TIME_WAIT. If a non-FIN packet is

received, the system restarts the timer upon receiving the last non-FIN packet. The connection is broken after the timer expires.

The actual length of the finwait timer is determined by the following formula:

Actual length of the finwait timer = (Configured length of the finwait timer – 75) + configured length of the synwait timer

To configure TCP timers:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Configure TCP timers.	<ul style="list-style-type: none"> Configure the TCP synwait timer: tcp timer syn-timeout <i>time-value</i> Configure the TCP finwait timer: tcp timer fin-timeout <i>time-value</i> 	<p>Optional.</p> <p>By default:</p> <ul style="list-style-type: none"> The synwait timer is 75 seconds. The finwait timer is 675 seconds.

Configuring ICMP to send error packets

Sending error packets is a major function of ICMP. In case of network abnormalities, error packets are usually sent by the network or transport layer protocols to notify corresponding devices so as to facilitate control and management.

Advantages of sending ICMP error packets

ICMP error packets include the following types:

- ICMP redirect packets

A host may have only a default route to the default gateway in its routing table after startup. If the following conditions are satisfied, the default gateway will send ICMP redirect packets to the source host, telling it to reselect a correct next hop to send the subsequent packets:

- The receiving and forwarding interfaces are the same.
- The selected route has not been created or modified by an ICMP redirect packet.
- The selected route is not the default route of the device.
- There is no source route option in the packet.

The ICMP redirect packets function simplifies host administration and enables a host to gradually establish a sound routing table to find the best route.

- ICMP timeout packets

If the device receives an IP packet with a timeout error, it drops the packet and sends an ICMP timeout packet to the source.

The device sends an ICMP timeout packet under the following conditions:

- If the device finds that the destination of a packet is not itself and the TTL field of the packet is 1, it will send a "TTL timeout" ICMP error message.
- When the device receives the first fragment of an IP datagram whose destination is the device itself, it starts a timer. If the timer times out before all the fragments of the datagram are received, the device will send a "reassembly timeout" ICMP error packet.

- ICMP destination unreachable packets

If the device receives an IP packet with the destination unreachable, it will drop the packet and send an ICMP destination unreachable error packet to the source.

Conditions for sending an ICMP destination unreachable packet:

- If neither a route nor the default route for forwarding a packet is available, the device will send a "network unreachable" ICMP error packet.
- If the destination of a packet is local but the transport layer protocol of the packet is not supported by the local device, the device sends a "protocol unreachable" ICMP error packet to the source.
- When receiving a packet with the destination being local and transport layer protocol being UDP, if the packet's port number does not match the running process, the device will send the source a "port unreachable" ICMP error packet.
- If the source uses "strict source routing" to send packets, but the intermediate device finds that the next hop specified by the source is not directly connected, the device will send the source a "source routing failure" ICMP error packet.
- When forwarding a packet, if the MTU of the sending interface is smaller than the packet, but the packet has been set as "Don't Fragment," the device will send the source a "fragmentation needed and Don't Fragment (DF-set)" ICMP error packet.

Disadvantages of sending ICMP error packets

Sending ICMP error packets facilitates network control and management, but it has the following disadvantages:

- Increases network traffic.
- A device's performance degrades if it receives a lot of malicious packets that cause it to respond with ICMP error packets.
- A host's performance degrades if the redirection function increases the size of its routing table.
- End users are affected because of receiving ICMP destination unreachable packets caused by malicious users.

To prevent such problems, disable the device from sending ICMP error packets.

Configuration procedure

The device stops sending "TTL timeout" ICMP error packets after sending ICMP timeout packets is disabled. However, "reassembly timeout" error packets will be sent normally.

To enable sending ICMP error packets:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable sending ICMP error packets.	<ul style="list-style-type: none"> • Enable sending ICMP redirect packets: ip redirects enable • Enable sending ICMP timeout packets: ip ttl-expires enable • Enable sending ICMP destination unreachable packets: ip unreachable enable 	Disabled by default

Displaying and maintaining IP performance optimization

Task	Command	Remarks
Display TCP connection statistics.	display tcp statistics [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display UDP statistics.	display udp statistics [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display statistics of IP packets .	display ip statistics [<i>slot slot-number</i>] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display ICMP statistics .	display icmp statistics [<i>slot slot-number</i>] [{ begin exclude include } <i>regular-expression</i>] <i>regular-expression</i>]	Available in any view
Display socket information .	display ip socket [socketype <i>sock-type</i>] [<i>task-id socket-id</i>] [<i>slot slot-number</i>] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display FIB information.	display fib [acl <i>acl-number</i> ip-prefix <i>ip-prefix-name</i>] [{ begin include exclude } <i>regular-expression</i>]	Available in any view
Display FIB information matching the specified destination IP address.	display fib <i>ip-address</i> [<i>mask</i> <i>mask-length</i>] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Clear statistics of IP packets .	reset ip statistics [<i>slot slot-number</i>]	Available in user view
Clear statistics of TCP connections.	reset tcp statistics	Available in user view
Clear statistics of UDP traffic.	reset udp statistics	Available in user view

Configuring UDP helper

Overview

UDP helper functions as a relay agent that converts UDP broadcast packets into unicast packets and forwards them to a specified destination server. This is helpful when a host cannot obtain network configuration information or request device names through broadcasting because the server or host to be requested is located on another broadcast domain.

With UDP helper enabled, a device decides whether to forward a received UDP broadcast packet according to the UDP destination port number of the packet.

- If the destination port number of the packet matches the one pre-configured on the device, the device modifies the destination IP address in the IP header, and then sends the packet to the specified destination server.
- If the destination port number of the packet does not match the one pre-configured on the device, the device sends the packet to the upper layer protocol for processing.

Configuration restrictions and guidelines

- The receiving of directed broadcasts to a directly connected network is disabled by default on the switch. As a result, UDP helper is available only when the **ip forward-broadcast** command is configured in system view. For more information about reception and forwarding of directed broadcasts to a directly connected network, see "Configuring IP performance optimization."
- A UDP helper enabled device must not forward DHCP broadcast packets that use destination port 67 or 68. Therefore, the UDP port numbers set with the **udp-helper port** command must not include 67 or 68.
- You can specify a port number or the corresponding parameter for a UDP port to forward packets. For example, **udp-helper port 53** and **udp-helper port dns** specify the same UDP port number.
- The configuration of all UDP ports is removed if you disable UDP helper.
- You can configure up to 256 UDP port numbers to enable the forwarding of packets with these UDP port numbers.
- You can configure up to 20 destination servers on an interface.

Configuration procedure

To configure UDP helper:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable UDP helper.	udp-helper enable	Disabled by default.
3. Enable the forwarding of packets with the specified UDP destination port numbers.	udp-helper port { <i>port-number</i> dns netbios-ds netbios-ns tacacs fttp time }	No UDP port number is specified by default.

Step	Command	Remarks
4.	Enter interface view. <code>interface interface-type interface-number</code>	N/A
5.	Specify the destination server to which UDP packets are to be forwarded. <code>udp-helper server ip-address</code>	No destination server is specified by default.

Displaying and maintaining UDP helper

Task	Command	Remarks
Displays information about forwarded UDP packets.	<code>display udp-helper server [interface interface-type interface-number] [{ begin exclude include } regular-expression]</code>	Available in any view
Clear statistics about packets forwarded.	<code>reset udp-helper packet</code>	Available in user view

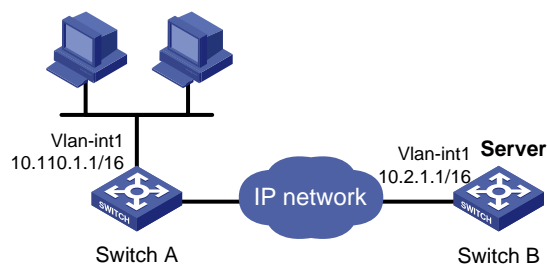
UDP helper configuration example

Network requirements

As shown in [Figure 49](#), the IP address of VLAN-interface 1 of Switch A is 10.110.1.1/16, and the interface connects to the subnet 10.110.0.0/16.

Configure UDP helper to forward broadcast packets with UDP destination port number 55 and destination IP address 255.255.255.255 or 10.110.255.255 to the destination server 10.2.1.1/16 in public network.

Figure 49 Network diagram



Configuration procedure

Verify that a route from Switch A to the subnet 10.2.0.0/16 is available.

Enable Switch A to receive directed broadcasts.

```

<SwitchA> system-view
[SwitchA] ip forward-broadcast
  
```

Enable UDP helper.

```

[SwitchA] udp-helper enable
  
```



```
# Enable the forwarding broadcast packets with the UDP destination port 55.
[SwitchA] udp-helper port 55

# Specify the destination server 10.2.1.1 on VLAN-interface 1 in public network.
[SwitchA] interface vlan-interface 1
[SwitchA-Vlan-interface1] ip address 10.110.1.1 16
[SwitchA-Vlan-interface1] udp-helper server 10.2.1.1
```

Configuring IPv6 basics

Overview

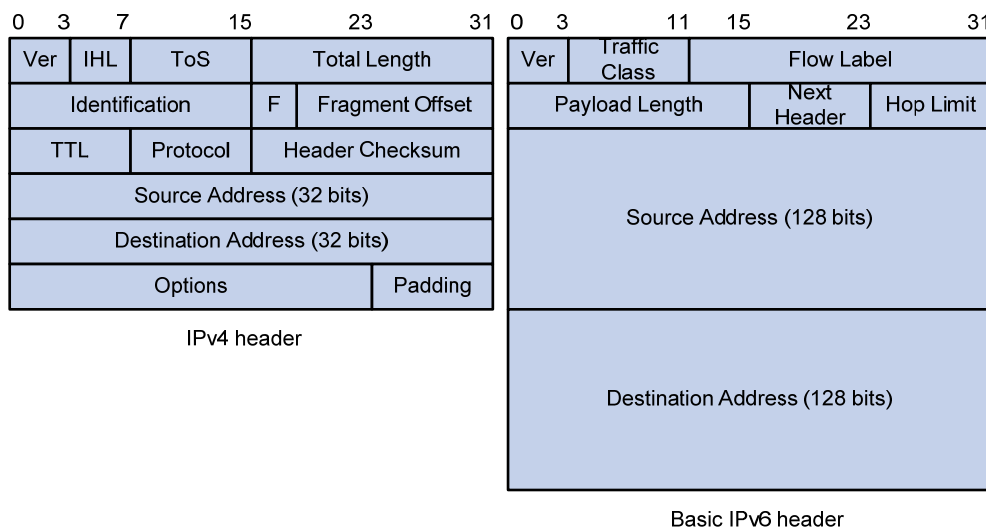
Internet Protocol Version 6 (IPv6), also called IP next generation (IPng), was designed by the Internet Engineering Task Force (IETF) as the successor to Internet Protocol version 4 (IPv4). The significant difference between IPv6 and IPv4 is that IPv6 increases the IP address size from 32 bits to 128 bits.

IPv6 features

Header format simplification

IPv6 removes several IPv4 header fields or moves them to the IPv6 extension headers to reduce the length of the basic IPv6 packet header. The basic IPv6 packet header has a fixed length of 40 bytes to simplify IPv6 packet handling and to improve forwarding efficiency. Although IPv6 address size is four times larger than IPv4 address size, the basic IPv6 packet header size is only twice the size of the option-less IPv4 packet header.

Figure 50 IPv4 packet header format and basic IPv6 packet header format



Larger address space

The source and destination IPv6 addresses are 128 bits (or 16 bytes) long. IPv6 can provide 3.4×10^{38} addresses to meet the requirements of hierarchical address division and the allocation of public and private addresses.

Hierarchical address structure

IPv6 uses hierarchical address structure to speed up route lookups and reduce the IPv6 routing table size through route aggregation.

Address autoconfiguration

To simplify host configuration, IPv6 supports stateful and stateless address autoconfiguration.

- Stateful address autoconfiguration enables a host to acquire an IPv6 address and other configuration information from a server (for example, a DHCP server).
- Stateless address autoconfiguration enables a host to automatically generate an IPv6 address and other configuration information by using its link-layer address and the prefix information advertised by a router.

To communicate with other hosts on the same link, a host automatically generates a link-local address based on its link-layer address and the link-local address prefix (FE80::/10).

Built-in security

IPv6 defines extension headers to support IPsec. IPsec provides end-to-end security for network security solutions and enhances interoperability among different IPv6 applications.

QoS support

The Flow Label field in the IPv6 header allows the device to label the packets and facilitates the special handling of a flow.

Enhanced neighbor discovery mechanism

The IPv6 neighbor discovery protocol is implemented through a group of Internet Control Message Protocol version 6 (ICMPv6) messages to manage the information exchange among neighboring nodes on the same link. The group of ICMPv6 messages replaces Address Resolution Protocol (ARP) messages, Internet Control Message Protocol version 4 (ICMPv4) Router Discovery messages, and ICMPv4 Redirect messages and provides a series of other functions.

Flexible extension headers

IPv6 eliminates the Options field in the header and introduces optional extension headers to provide scalability and improve efficiency. The Options field in the IPv4 packet header contains a maximum of 40 bytes, whereas the IPv6 extension headers are restricted to the maximum size of IPv6 packets only.

IPv6 addresses

IPv6 address format

An IPv6 address is represented as a set of 16-bit hexadecimal numbers separated by colons. An IPv6 address is divided into eight groups, and each 16-bit group is represented by four hexadecimal numbers, for example, 2001:0000:130F:0000:0000:09C0:876A:130B.

To simplify the representation of IPv6 addresses, you can handle zeros in IPv6 addresses by using the following methods:

- The leading zeros in each group can be removed. For example, the previous address can be represented in a shorter format as 2001:0:130F:0:0:9C0:876A:130B.
- If an IPv6 address contains two or more consecutive groups of zeros, they can be replaced by a double colon (::). For example, the previous address can be represented in the shortest format as 2001:0:130F::9C0:876A:130B.

A double colon may appear once or not at all in an IPv6 address. This limit allows the device to determine how many zeros the double colon represents, and correctly convert it to zeros to restore a 128-bit IPv6 address.

An IPv6 address consists of an address prefix and an interface ID, both of which are equivalent to the network ID and the host ID of an IPv4 address, respectively.

An IPv6 address prefix is written in IPv6-address/prefix-length notation where the IPv6-address is represented in any of the formats previously mentioned and the prefix-length is a decimal number indicating how many leftmost bits of the IPv6 address comprises the address prefix.

IPv6 address types

IPv6 addresses fall into the following types:

- **Unicast address**—An identifier for a single interface, similar to an IPv4 unicast address. A packet sent to a unicast address is delivered to the interface identified by that address.
- **Multicast address**—An identifier for a set of interfaces (typically belonging to different nodes), similar to an IPv4 multicast address. A packet sent to a multicast address is delivered to all interfaces identified by that address.
- **Anycast address**—An identifier for a set of interfaces (typically belonging to different nodes). A packet sent to an anycast address is delivered to the nearest one of the interfaces identified by that address. The nearest interface is chosen according to the routing protocols' measure of distance.

NOTE:

There are no broadcast addresses in IPv6. Their function is replaced by multicast addresses.

The type of an IPv6 address is designated by the first several bits, the format prefix. [Table 5](#) lists the mappings between address types and format prefixes.

Table 5 Mappings between address types and format prefixes

Type	Format prefix (binary)	IPv6 prefix ID	
Unicast address	Unspecified address	00...0 (128 bits)	::/128
	Loopback address	00...1 (128 bits)	::1/128
	Link-local address	1111111010	FE80::/10
	Site-local address	1111111011	FECO::/10
	Global unicast address	Other forms	N/A
Multicast address	11111111	FF00::/8	
Anycast address	Anycast addresses use the unicast address space and have the identical structure of unicast addresses.		

Unicast addresses

Unicast addresses comprise global unicast addresses, link-local unicast addresses, site-local unicast addresses, the loopback address, and the unspecified address.

- Global unicast addresses, equivalent to public IPv4 addresses, are provided for network service providers. This type of address allows efficient prefix aggregation to restrict the number of global routing entries.
- Link-local addresses are used for communication among link-local nodes for neighbor discovery and stateless autoconfiguration. Packets with link-local source or destination addresses are not forwarded to other links.
- Site-local unicast addresses are similar to private IPv4 addresses. Packets with site-local source or destination addresses are not forwarded out of the local site (or a private network).

- A loopback address is 0:0:0:0:0:0:0:1 (or ::1). It cannot be assigned to any physical interface and can be used by a node to send an IPv6 packet to itself in the same way as the loopback address in IPv4.
- An unspecified address is 0:0:0:0:0:0:0:0 (or ::). It cannot be assigned to any node. Before acquiring a valid IPv6 address, a node fills this address in the source address field of IPv6 packets. The unspecified address cannot be used as a destination IPv6 address.

Multicast addresses

IPv6 multicast addresses listed in [Table 6](#) are reserved for special purposes.

Table 6 Reserved IPv6 multicast addresses

Address	Application
FF01::1	Node-local scope all-nodes multicast address
FF02::1	Link-local scope all-nodes multicast address
FF01::2	Node-local scope all-routers multicast address
FF02::2	Link-local scope all-routers multicast address

Multicast addresses also include solicited-node addresses. A node uses a solicited-node multicast address to acquire the link-layer address of a neighboring node on the same link and to detect duplicate addresses. Each IPv6 unicast or anycast address has a corresponding solicited-node address. The format of a solicited-node multicast address is: FF02:0:0:0:0:1:FFXX:XXXX where FF02:0:0:0:0:1:FF is fixed and consists of 104 bits, and XX:XXXX is the last 24 bits of an IPv6 unicast address or anycast address.

EUI-64 address-based interface identifiers

An interface identifier is 64 bits and uniquely identifies an interface on a link.

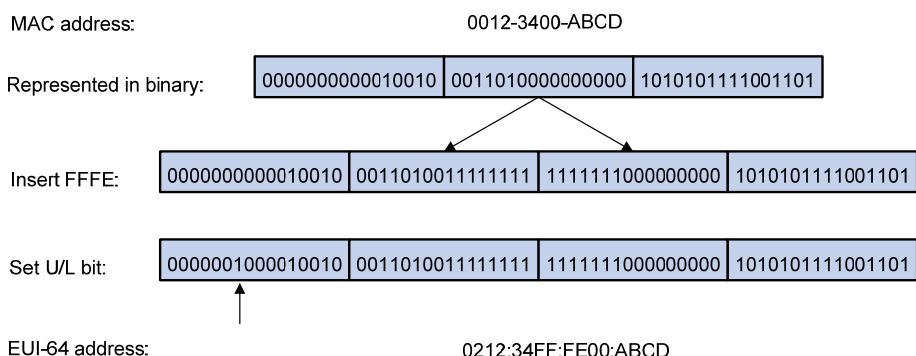
Interfaces generate EUI-64 address-based interface identifiers differently.

- On an IEEE 802 interface (such as a VLAN interface)

The interface identifier is derived from the link-layer address (typically a MAC address) of the interface. The MAC address is 48 bits long. To obtain an EUI-64 address-based interface identifier, you must insert the hexadecimal number FFFE (16 bits of 1111111111111110) into the MAC address (behind the 24th high-order bit), and set the universal/local (U/L) bit (which is the seventh high-order bit) to 1, to make sure that the obtained EUI-64 address-based interface identifier is globally unique.

[Figure 51](#) shows how an EUI-64 address-based interface identifier is generated from a MAC address.

Figure 51 Converting a MAC address into an EUI-64 address-based interface identifier



- On a tunnel interface
The lower 32 bits of the EUI-64 address-based interface identifier are the source IPv4 address of the tunnel interface. The higher 32 bits of the EUI-64 address-based interface identifier of an ISATAP tunnel interface are 0000:5EFE, whereas those of other tunnel interfaces are all zeros.
- On an interface of another type
The EUI-64 address-based interface identifier is generated randomly by the device.

IPv6 neighbor discovery protocol

The IPv6 Neighbor Discovery (ND) protocol uses five types of ICMPv6 messages to implement the following functions:

- [Address resolution](#)
- [Neighbor reachability detection](#)
- [Duplicate address detection](#)
- [Router/prefix discovery and address autoconfiguration](#)
- [Redirection](#)

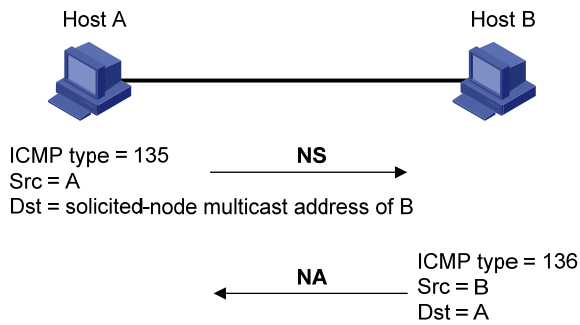
Table 7 ICMPv6 messages used by ND

ICMPv6 message	Type	Function
Neighbor Solicitation (NS) message	135	Acquires the link-layer address of a neighbor.
		Verifies whether a neighbor is reachable.
		Detects duplicate addresses.
Neighbor Advertisement (NA) message	136	Responds to an NS message.
		Notifies the neighboring nodes of link layer changes.
Router Solicitation (RS) message	133	Requests an address prefix and other configuration information for autoconfiguration after startup.
Router Advertisement (RA) message	134	Responds to an RS message.
		Advertises information such as the Prefix Information options and flag bits.
Redirect message	137	Informs the source host of a better next hop on the path to a particular destination when certain conditions are satisfied.

Address resolution

This function is similar to the ARP function in IPv4. An IPv6 node acquires the link-layer addresses of neighboring nodes on the same link through NS and NA message exchanges. [Figure 52](#) shows how Host A acquires the link-layer address of Host B on a single link.

Figure 52 Address resolution



The address resolution operates in the following steps:

1. Host A multicasts an NS message. The source address of the NS message is the IPv6 address of the sending interface of Host A and the destination address is the solicited-node multicast address of Host B. The NS message contains the link-layer address of Host A.
2. After receiving the NS message, Host B determines whether the destination address of the packet is its solicited-node multicast address. If yes, Host B learns the link-layer address of Host A, and then unicasts an NA message containing its link-layer address.
3. Host A acquires the link-layer address of Host B from the NA message.

Neighbor reachability detection

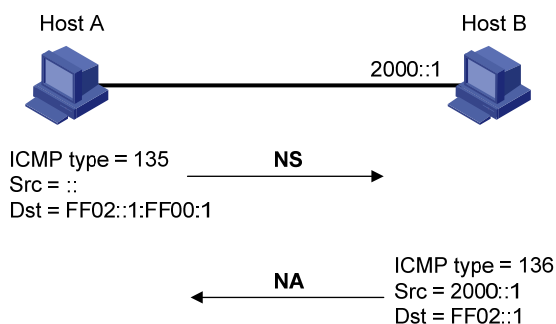
After Host A acquires the link-layer address of its neighbor Host B, Host A can use NS and NA messages to check whether Host B is reachable.

1. Host A sends an NS message whose destination address is the IPv6 address of Host B.
2. If Host A receives an NA message from Host B, Host A decides that Host B is reachable. Otherwise, Host B is unreachable.

Duplicate address detection

After Host A acquires an IPv6 address, it performs Duplicate Address Detection (DAD) to check whether the address is being used by any other node (similar to the gratuitous ARP function in IPv4). DAD is accomplished through NS and NA message exchanges. Figure 53 shows the DAD process.

Figure 53 Duplicate address detection



1. Host A sends an NS message whose source address is the unspecified address and whose destination address is the corresponding solicited-node multicast address of the IPv6 address to be detected. The NS message contains the IPv6 address.
2. If Host B uses this IPv6 address, Host B returns an NA message. The NA message contains the IPv6 address of Host B.

3. Host A learns that the IPv6 address is being used by Host B after receiving the NA message from Host B. If receiving no NA message, Host A decides that the IPv6 address is not in use and uses this address.

Router/prefix discovery and address autoconfiguration

Router/prefix discovery enables a node to locate the neighboring routers and to learn from the received RA message configuration parameters such as the prefix of the network where the node is located.

Stateless address autoconfiguration enables a node to generate an IPv6 address automatically according to the information obtained through router/prefix discovery.

Router/prefix discovery is implemented through RS and RA messages in the following steps:

1. At startup, a node sends an RS message to request the address prefix and other configuration information for autoconfiguration.
2. A router returns an RA message containing information such as Prefix Information options. (The router also periodically sends an RA message. In addition to an address prefix, the Prefix Information option also contains the preferred lifetime and valid lifetime of the address prefix. Nodes update the preferred lifetime and valid lifetime accordingly through periodic RA messages.)
3. The node automatically generates an IPv6 address and other configuration information according to the address prefix and other configuration parameters in the RA message. (The automatically generated address is applicable within the valid lifetime and is removed when the valid lifetime expires.)

Redirection

A newly started host may contain only a default route to the gateway in its routing table. When certain conditions are satisfied, the gateway sends an ICMPv6 Redirect message to the source host, so the host can select a better next hop to forward packets (similar to the ICMP redirection function in IPv4).

The gateway sends an ICMPv6 Redirect message when the following conditions are satisfied.

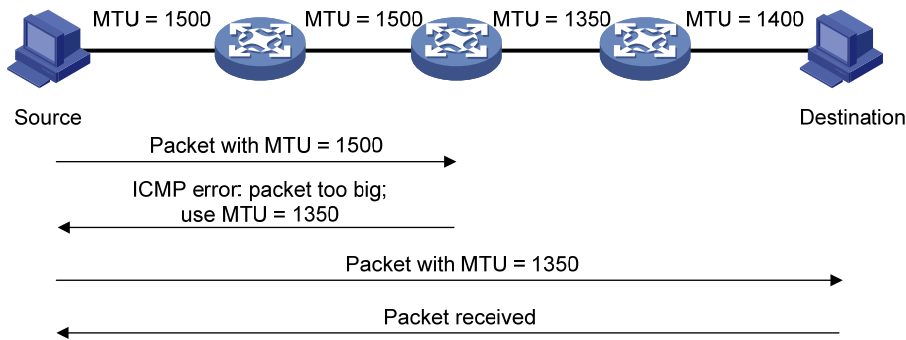
- The receiving interface is the forwarding interface.
- The selected route itself is not created or modified by an ICMPv6 Redirect message.
- The selected route is not the default route.

IPv6 path MTU discovery

The links that a packet passes from a source to a destination may have different MTUs. In IPv6, when the packet size exceeds the path MTU of a link, the packet is fragmented at the source end of the link to reduce the processing pressure on intermediate devices and to use network resources effectively.

The path MTU discovery mechanism is designed to find the minimum MTU of all links in the path between a source and a destination. [Figure 54](#) shows how a source host discovers the path MTU to a destination host.

Figure 54 Path MTU discovery process



1. The source host compares its MTU with the packet to be sent, performs necessary fragmentation, and sends the resulting packet to the destination host.
2. If the MTU supported by a forwarding interface is smaller than the packet, the device discards the packet and returns an ICMPv6 error packet containing the interface MTU to the source host.
3. After receiving the ICMPv6 error packet, the source host uses the returned MTU to limit the packet size, performs fragmentation, and sends the resulting packet to the destination host.
4. Step 2 and step 3 are repeated until the destination host receives the packet. In this way, the source host decides the minimum MTU of all links in the path to the destination host.

IPv6 transition technologies

Before IPv6 dominates the Internet, high-efficient and seamless IPv6 transition technologies are needed to enable communication between IPv4 and IPv6 networks. Several IPv6 transition technologies can be used in different environments and periods, such as dual stack (RFC 2893) and tunneling (RFC 2893).

Dual stack

Dual stack is the most direct transition approach. A network node that supports both IPv4 and IPv6 is a dual stack node. A dual stack node configured with an IPv4 address and an IPv6 address can forward both IPv4 and IPv6 packets. For an upper layer application that supports both IPv4 and IPv6, either TCP or UDP can be selected at the transport layer, whereas the IPv6 stack is preferred at the network layer. Dual stack is suitable for communication between IPv4 nodes or between IPv6 nodes. It is the basis of all transition technologies. However, it does not solve the IPv4 address depletion issue because each dual stack node must have a globally unique IP address.

Tunneling

Tunneling is an encapsulation technology that utilizes one network protocol to encapsulate packets of another network protocol and transfer them over the network.

Protocols and standards

Protocols and standards related to IPv6 include:

- RFC 1881, *IPv6 Address Allocation Management*
- RFC 1887, *An Architecture for IPv6 Unicast Address Allocation*
- RFC 1981, *Path MTU Discovery for IP version 6*
- RFC 2375, *IPv6 Multicast Address Assignments*
- RFC 2460, *Internet Protocol, Version 6 (IPv6) Specification*

- RFC 2464, *Transmission of IPv6 Packets over Ethernet Networks*
- RFC 2526, *Reserved IPv6 Subnet Anycast Addresses*
- RFC 2894, *Router Renumbering for IPv6*
- RFC 3307, *Allocation Guidelines for IPv6 Multicast Addresses*
- RFC 3513, *Internet Protocol Version 6 (IPv6) Addressing Architecture*
- RFC 4191, *Default Router Preferences and More-Specific Routes*
- RFC 4443, *Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification*
- RFC 4861, *Neighbor Discovery for IP Version 6 (IPv6)*
- RFC 4862, *IPv6 Stateless Address Autoconfiguration*

IPv6 basics configuration task list

Task	Remarks	
Configuring basic IPv6 functions	Enabling IPv6	Required
	Configuring an IPv6 global unicast address	Required to configure one
	Configuring an IPv6 link-local address	
	Configure an IPv6 anycast address	
Configuring IPv6 ND	Configuring a static neighbor entry	Optional
	Configuring the maximum number of neighbors dynamically learned	Optional
	Setting the age timer for ND entries in stale state	Optional
	Configuring parameters related to RA messages	Optional
	Configuring the maximum number of attempts to send an NS message for DAD	Optional
	Configuring ND snooping	Optional
	Enabling ND proxy	Optional
Configuring path MTU discovery	Configuring a static path MTU for a specific IPv6 address	Optional
	Configuring the aging time for dynamic path MTUs	Optional
Configuring IPv6 TCP properties		Optional
Configuring ICMPv6 packet sending	Configuring the maximum ICMPv6 error packets sent in an interval	Optional
	Enabling replying to multicast echo requests	Optional
	Enabling sending ICMPv6 time exceeded messages	Optional
	Enabling sending ICMPv6 destination unreachable messages	Optional

Configuring basic IPv6 functions

Enabling IPv6

Enable IPv6 before you perform any IPv6-related configuration. Without IPv6 enabled, an interface cannot forward IPv6 packets even if it has an IPv6 address configured.

To enable IPv6:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable IPv6.	ipv6	Disabled by default

Configuring an IPv6 global unicast address

Configure an IPv6 global unicast address by using the following options:

- **EUI-64 IPv6 addressing**—The IPv6 address prefix of an interface is manually configured, and the interface identifier is generated automatically by the interface.
- **Manual configuration**—The IPv6 global unicast address is configured manually.
- **Stateless address autoconfiguration**—The IPv6 global unicast address is generated automatically based on the address prefix information contained in the RA message.

Follow these guidelines when you configure an IPv6 global unicast address:

- You can configure multiple IPv6 global unicast addresses with different prefixes on an interface.
- A manually configured global unicast address takes precedence over an automatically generated one. If a global unicast address has been automatically generated on an interface when you manually configure another one with the same address prefix, the latter overwrites the previous. The overwritten automatic global unicast address will not be restored even if the manual one is removed. Instead, a new global unicast address will be automatically generated based on the address prefix information in the RA message that the interface receives at the next time.

EUI-64 IPv6 addressing

To configure an interface to generate an EUI-64 IPv6 address:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Configure the interface to generate an EUI-64 IPv6 address.	ipv6 address <i>ipv6-address/prefix-length</i> eui-64	By default, no IPv6 global unicast address is configured on an interface.

Manual configuration

To specify an IPv6 address manually for an interface:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Configure an IPv6 address manually.	ipv6 address { <i>ipv6-address</i> <i>prefix-length</i> <i>ipv6-address/prefix-length</i> }	By default, no IPv6 global unicast address is configured on an interface.

Stateless address autoconfiguration

To configure an interface to generate an IPv6 address by using stateless address autoconfiguration:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Configure an IPv6 address to be generated through stateless address autoconfiguration.	ipv6 address auto	By default, no IPv6 global unicast address is configured on an interface.

NOTE:

Using the **undo ipv6 address auto** command on an interface removes all IPv6 global unicast addresses automatically generated on the interface.

With stateless address autoconfiguration enabled on an interface, the device automatically generates an IPv6 global unicast address by using the address prefix information in the received RA message and the interface ID. On an IEEE 802 interface (such as a VLAN interface), the interface ID is generated based on the MAC address of the interface, and is globally unique. As a result, the interface ID portion of the IPv6 global address remains unchanged and exposes the sender. An attacker can further exploit communication details such as the communication peer and time.

To fix the vulnerability, configure the temporary address function that enables the system to generate and use temporary IPv6 addresses with different interface ID portions on an interface. With this function configured on an IEEE 802 interface, the system can generate two addresses, public IPv6 address and temporary IPv6 address.

- **Public IPv6 address**—Comprises an address prefix provided by the RA message, and a fixed interface ID generated based on the MAC address of the interface.
- **Temporary IPv6 address**—Comprises an address prefix provided by the RA message, and a random interface ID generated through MD5.

Before sending a packet, the system preferably uses the temporary IPv6 address of the sending interface as the source address of the packet to be sent. When this temporary IPv6 address expires, the system removes it and generates a new one. This enables the system to send packets with different source addresses through the same interface. If the temporary IPv6 address cannot be used because of a DAD conflict, the public IPv6 address is used.

The preferred lifetime and valid lifetime for temporary IPv6 addresses are specified as follows:

- The preferred lifetime of a temporary IPv6 address takes the value of the smaller of the following values:
 - The preferred lifetime of the address prefix in the RA message.

- The preferred lifetime configured for temporary IPv6 addresses minus DESYNC_FACTOR (which is a random number ranging 0 to 600, in seconds).
- The valid lifetime of a temporary IPv6 address takes the value of the smaller of the following values:
 - The valid lifetime of the address prefix.
 - The valid lifetime configured for temporary IPv6 addresses.

To configure the temporary address function:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Configure the system to generate and preferably use the temporary IPv6 address of the sending interface as the source address of the packet to be sent.	ipv6 prefer temporary-address [<i>valid-lifetime preferred-lifetime</i>]	By default, the system does not generate or use a temporary IPv6 address.

You must also enable stateless address autoconfiguration on an interface if you need temporary IPv6 addresses to be generated on that interface. Temporary IPv6 addresses do not override public IPv6 addresses. Therefore, an interface may have multiple IPv6 addresses with the same address prefix but different interface ID portions.

If the public IPv6 address fails to be generated on an interface because of a prefix conflict or other reasons, no temporary IPv6 address will be generated on the interface.

Configuring an IPv6 link-local address

IPv6 link-local addresses can be configured in either of the following ways:

- **Automatic generation**—The device automatically generates a link-local address for an interface according to the link-local address prefix (FE80::/10) and the link-layer address of the interface.
- **Manual assignment**—IPv6 link-local addresses can be assigned manually.

An interface can have only one link-local address. To avoid link-local address conflicts, use the automatic generation method.

Manual assignment takes precedence over automatic generation.

- If you first use automatic generation and then manual assignment, the manually assigned link-local address will overwrite the automatically generated one.
- If you first use manual assignment and then automatic generation, the automatically generated link-local address will not take effect and the link-local address is still the manually assigned one. If you delete the manually assigned address, the automatically generated link-local address is validated.

To configure automatic generation of an IPv6 link-local address for an interface:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A

Step	Command	Remarks
3.	Configure the interface to automatically generate an IPv6 link-local address. ipv6 address auto link-local	Optional. By default, no link-local address is configured on an interface. After an IPv6 global unicast address is configured on the interface, a link-local address is generated automatically.

To configure an IPv6 link-local address manually:

Step	Command	Remarks
1.	Enter system view. system-view	N/A
2.	Enter interface view. interface <i>interface-type</i> <i>interface-number</i>	N/A
3.	Configure an IPv6 link-local address manually. ipv6 address <i>ipv6-address</i> link-local	Optional. By default, no link-local address is configured on an interface. After an IPv6 global unicast address is configured on the interface, a link-local address is generated automatically.

After an IPv6 global unicast address is configured for an interface, a link-local address is generated automatically.

- The automatically generated link-local address is the same as the one generated by using the **ipv6 address auto link-local** command.
- If a link-local address is manually assigned to an interface, this manual link-local address takes effect. If the manually assigned link-local address is removed, the automatically generated link-local address takes effect.

The **undo ipv6 address auto link-local** command can only remove the link-local addresses generated through the **ipv6 address auto link-local** command.

- If an IPv6 global unicast address is already configured for an interface, the interface still has a link-local address because the system automatically generates one for the interface.
- If no IPv6 global unicast address is configured, the interface has no link-local address.

Configure an IPv6 anycast address

To configure an IPv6 anycast address for an interface:

Step	Command	Remarks
1.	Enter system view. system-view	N/A
2.	Enter interface view. interface <i>interface-type</i> <i>interface-number</i>	N/A
3.	Configure an IPv6 anycast address. ipv6 address <i>ipv6-address/prefix-length</i> anycast	Optional. By default, no IPv6 anycast address is configured on an interface.

Configuring IPv6 ND

Configuring a static neighbor entry

The IPv6 address of a neighboring node can be resolved into a link-layer address dynamically through NS and NA messages or through a manually configured static neighbor entry.

The device uniquely identifies a static neighbor entry by the neighbor's IPv6 address and the local Layer 3 interface number. You can configure a static neighbor entry by using either of the following methods:

- **Method 1**—Associate a neighbor IPv6 address and link-layer address with the Layer 3 interface of the local node.
- **Method 2**—Associate a neighbor IPv6 address and link-layer address with a port in a VLAN containing the local node.

You can use either of the previous configuration methods to configure a static neighbor entry for a VLAN interface.

- After a static neighbor entry is configured by using the first method, the device must resolve the corresponding Layer 2 port information about the VLAN interface.
- If you use the second method, make sure that the corresponding VLAN interface exists and that the Layer 2 port specified by *port-type port-number* belongs to the VLAN specified by *vlan-id*. After a static neighbor entry is configured, the device associates the VLAN interface with the IPv6 address to identify the static neighbor entry uniquely.

To configure a static neighbor entry:

Step	Command
1. Enter system view.	system-view
2. Configure a static neighbor entry.	ipv6 neighbor <i>ipv6-address mac-address</i> { <i>vlan-id port-type port-number</i> interface <i>interface-type interface-number</i> }

Configuring the maximum number of neighbors dynamically learned

The device can dynamically acquire the link-layer address of a neighboring node through NS and NA messages and add it into the neighbor table. A large table can reduce the forwarding performance of the device. You can restrict the size of the neighbor table by setting the maximum number of neighbors that an interface can dynamically learn. When the number of dynamically learned neighbors reaches the threshold, the interface will stop learning neighbor information.

To configure the maximum number of neighbors dynamically learned:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type interface-number</i>	N/A

Step	Command	Remarks
3.	Configure the maximum number of neighbors dynamically learned by an interface.	Optional. By default, a Layer 2 interface does not limit the number of neighbors dynamically learned, and a Layer 3 interface can learn up to 512 neighbors dynamically.

Setting the age timer for ND entries in stale state

ND entries in stale state have an age timer. If an ND entry in stale state is not refreshed before the timer expires, it transits to the delay state. If it is still not refreshed in five seconds, the ND entry transits to the probe state, and the device sends an NS message for detection. If no response is received, the device removes the ND entry.

To set the age timer for ND entries in stale state:

Step	Command	Remarks
1.	Enter system view.	N/A
2.	Set the age timer for ND entries in stale state.	Optional. Four hours by default.

Configuring parameters related to RA messages

You can enable an interface to send RA messages, and configure the interval for sending RA messages and parameters in RA messages. After receiving an RA message, a host can use these parameters to perform corresponding operations. [Table 8](#) lists and describes the configurable parameters in an RA message.

The maximum interval for sending RA messages should be less than (or equal to) the router lifetime in RA messages, so the router can be updated through an RA message before expiration.

The values of the NS retransmission timer and the reachable time configured for an interface are sent to hosts via RA messages. Furthermore, this interface sends NS messages at the interval of the NS retransmission timer and considers a neighbor reachable within the reachable time.

Table 8 Parameters in an RA message and their descriptions

Parameters	Description
Cur Hop Limit	When sending an IPv6 packet, a host uses the value to fill the Hop Limit field in IPv6 headers. The value is also filled into the Hop Limit field in the response packet of a device.
Prefix Information options	After receiving the prefix information, the hosts on the same link can perform stateless autoconfiguration.
MTU	Make sure that all nodes on a link use the same MTU value.

Parameters	Description
M flag	Determines whether hosts use the stateful autoconfiguration to acquire IPv6 addresses. If the M flag is set to 1, hosts use the stateful autoconfiguration (for example, through a DHCP server) to acquire IPv6 addresses. Otherwise, hosts use the stateless autoconfiguration to acquire IPv6 addresses and generate IPv6 addresses according to their own link-layer addresses and the obtained prefix information.
O flag	Determines whether hosts use stateful autoconfiguration to acquire other configuration information. If the O flag is set to 1, hosts use stateful autoconfiguration (for example, through a DHCP server) to acquire other configuration information. Otherwise, hosts use stateless autoconfiguration to acquire other configuration information.
Router Lifetime	Tells the receiving hosts how long the advertising device can live.
Retrans Timer	If the device fails to receive a response message within the specified time after sending an NS message, it will retransmit the NS message.
Reachable Time	If the neighbor reachability detection shows that a neighbor is reachable, the device considers the neighbor reachable within the specified reachable time. If the device must send a packet to the neighbor after the specified reachable time expires, the device will reconfirm whether the neighbor is reachable.

To allow sending of RA messages:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Disable RA message suppression.	undo ipv6 nd ra halt	By default, RA messages are suppressed.
4. Configure the maximum and minimum intervals for sending RA messages.	ipv6 nd ra interval <i>max-interval-value</i> <i>min-interval-value</i>	Optional. By default, the maximum interval for sending RA messages is 600 seconds, and the minimum interval is 200 seconds. The device sends RA messages at random intervals between the maximum interval and the minimum interval. The minimum interval should be less than or equal to 0.75 times the maximum interval.

To configure parameters related to RA messages:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Configure the hop limit.	ipv6 nd hop-limit <i>value</i>	Optional. 64 by default.
3. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A

Step	Command	Remarks
4. Configure the prefix information in RA messages.	ipv6 nd ra prefix { <i>ipv6-prefix</i> <i>prefix-length</i> <i>ipv6-prefix/prefix-length</i> } <i>valid-lifetime preferred-lifetime</i> [no-autoconfig off-link] *	Optional. By default, no prefix information is configured for RA messages, and the IPv6 address of the interface sending RA messages is used as the prefix information with valid lifetime 2592000 seconds (30 days) and preferred lifetime 604800 seconds (seven days).
5. Turn off the MTU option in RA messages.	ipv6 nd ra no-advlinkmtu	Optional. By default, RA messages contain the MTU option.
6. Set the M flag bit to 1.	ipv6 nd autoconfig managed-address-flag	Optional. By default, the M flag bit is set to 0 and hosts acquire IPv6 addresses through stateless autoconfiguration.
7. Set the O flag bit to 1.	ipv6 nd autoconfig other-flag	Optional. By default, the O flag bit is set to 0 and hosts acquire other configuration information through stateless autoconfiguration.
8. Configure the router lifetime in RA messages.	ipv6 nd ra router-lifetime <i>value</i>	Optional. 1800 seconds by default.
9. Set the NS retransmission timer.	ipv6 nd ns retrans-timer <i>value</i>	Optional. By default, the local interface sends NS messages at 1000 millisecond intervals, and the value of the Retrans Timer field in RA messages sent by the local interface is 0. The interval for retransmitting an NS message is determined by the receiving device.
10. Set the reachable time.	ipv6 nd nud reachable-time <i>value</i>	Optional. By default, the neighbor reachable time on the local interface is 30000 milliseconds, and the value of the Reachable Time field in the RA messages sent by the local interface is 0. The neighbor reachable time is determined by the receiving device.

Configuring the maximum number of attempts to send an NS message for DAD

An interface sends an NS message for DAD after acquiring an IPv6 address. If the interface does not receive a response within a specific time (determined by the **ipv6 nd ns retrans-timer** command), it continues to send an NS message. If the interface still does not receive a response after the number of sent attempts reaches the threshold (specified with the **ipv6 nd dad attempts** command), the acquired address is considered usable.

To configure the attempts to send an NS message for DAD:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Configure the number of attempts to send an NS message for DAD.	ipv6 nd dad attempts <i>value</i>	Optional. 1 by default. When the <i>value</i> argument is set to 0, DAD is disabled.

Configuring ND snooping

Introduction

The ND snooping feature is used in Layer 2 switching networks. It creates ND snooping entries using DAD NS messages.

ND snooping entries are used to do the following:

- Cooperate with the ND detection function. For more information about ND detection, see *Security Configuration Guide*.
- Cooperate with the IP Source Guard function. For more information about IP source guard, see *Security Configuration Guide*.
- Work in all SAVI scenarios. For more information about SAVI, see *Security Configuration Guide*.

After you enable ND snooping on a VLAN of a device, ND packets received by the interfaces of the VLAN are redirected to the CPU. When ND snooping is enabled globally, the CPU uses the ND packets to create or update ND snooping entries comprising source IPv6 address, source MAC address, receiving VLAN, and receiving port information.

The following items describe how an ND snooping entry is created, updated, and aged out.

1. Create an ND snooping entry.

The device only uses received DAD NS messages to create ND snooping entries.

2. Update an ND snooping entry.

Upon receiving an ND packet, the device searches the ND snooping table for an entry containing the source IPv6 address of the packet. If the entry was refreshed within one second, the device does not update the entry. If the entry is not refreshed for more than one second, the device matches the MAC address of the ND packet and the receiving port against that in the entry.

- If both of them match those in the entry, the device updates the aging time of the ND snooping entry.
- If neither of them matches the entry and the received packet is a DAD NS message, the message is ignored.
- If neither of them matches the entry and the received packet is not a DAD NS message, the device performs active acknowledgement.

The active acknowledgement is performed in the following steps.

- The device checks the validity of the existing ND snooping entry. The device sends out a DAD NS message including the IPv6 address of the ND snooping entry. If a corresponding NA message (whose source IPv6 address, source MAC address, receiving port, and source VLAN

are consistent with those of the existing entry) is received, the device updates the aging time of the existing entry. If no corresponding NA message is received within one second after the DAD NS message is sent, the device starts to check the validity of the received ND packet.

- To check the validity of the received ND packet (packet A for example), the device sends out a DAD NS message including the source IPv6 address of packet A. If a corresponding NA message (whose source IPv6 address, source MAC address, receiving port, and source VLAN are consistent with those of packet A) is received, the device updates the aging time of the entry. If no corresponding NA message is received within one second after the DAD NS message is sent, the device does not update the entry.

3. Age out an ND snooping entry.

An ND snooping entry is aged out after 25 minutes. If an ND snooping entry is not updated within 15 minutes, the device performs active acknowledgement.

The device sends out a DAD NS message including the IPv6 address of the ND snooping.

- If a corresponding NA message is received (the source IPv6 address, source MAC address, receiving port, and source VLAN are consistent with those of the existing entry), the device updates the aging time of the existing entry.
- If no corresponding NA message is received within one second after the DAD NS message is sent out, the device removes the entry when the timer expires.

Configuration procedure

To configure ND snooping:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Configure ND snooping.	<ul style="list-style-type: none"> • Enable ND snooping based on global unicast addresses (the devices use DAD NS messages containing global unicast addresses to create ND snooping entries): ipv6 nd snooping enable global • Enable ND snooping based on link local addresses (the devices use DAD NS messages containing link local addresses to create ND snooping entries): ipv6 nd snooping enable link-local 	<p>Use either approach.</p> <p>By default, ND snooping is disabled.</p>
3. Enter VLAN view.	vlan <i>vlan-id</i>	N/A
4. Enable ND snooping.	ipv6 nd snooping enable	Disabled by default.
5. Return to system view.	quit	N/A
6. Enter Layer 2 Ethernet port view/Layer 2 aggregate interface view.	interface <i>interface-type interface-number</i>	N/A
7. Configure the maximum number of ND snooping entries the interface can learn.	ipv6 nd snooping max-learning-num <i>number</i>	<p>Optional.</p> <p>By default, the number of ND snooping entries an interface can learn is unlimited.</p>

Step	Command	Remarks
8. Configure the interface as an uplink interface and disable it from learning ND snooping entries.	ipv6 nd snooping uplink	Optional. By default, when ND snooping is enabled on the device, an interface is allowed to learn ND snooping entries.

Enabling ND proxy

ND proxy supports the NS and NA messages only.

Introduction

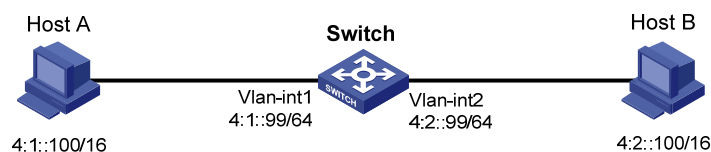
If a host sends an NS message requesting the hardware address of another host that is isolated from the sending host at Layer 2, the device between the hosts must be able to forward the NS message to allow Layer 3 communication between the two hosts. This process is achieved by ND proxy.

Depending on application scenarios, ND proxy falls into common ND proxy and local ND proxy.

- Common ND proxy

As shown in [Figure 55](#), VLAN-interface 1 with IPv6 address 4:1::99/64 and VLAN-interface 2 with IPv6 address 4:2::99/64 belong to different subnets. Host A and Host B reside on the same network but in different broadcast domains.

Figure 55 Application environment of common ND proxy



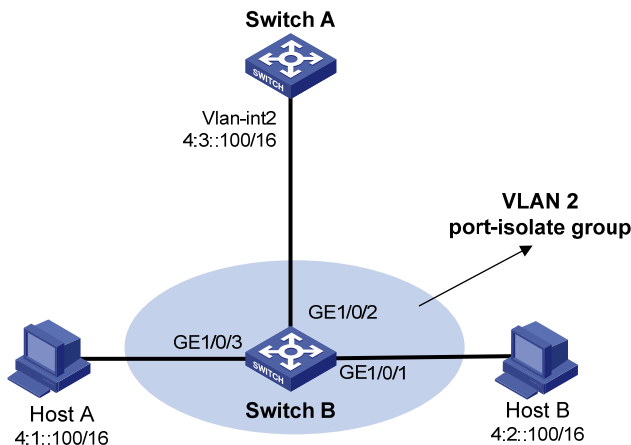
Because Host A's IPv6 address is on the same subnet as Host B's, Host A directly sends an NS message to obtain Host B's MAC address. However, Host B cannot receive the NS message because they belong to different broadcast domains.

To solve this problem, enable common ND proxy on VLAN-interface 1 and VLAN-interface 2 of the switch. The switch finds the matching forwarding entry according to the destination IPv6 address of the NS message and sends the message through the output interface of that entry. Upon receiving the NS message, Host B sends an NA message to the switch, which forwards it to Host A.

- Local ND proxy

As shown in [Figure 56](#), both Host A and Host B belong to VLAN 2, but they connect to GigabitEthernet 1/0/3 and GigabitEthernet 1/0/1 respectively, which are isolated at Layer 2.

Figure 56 Application environment of local ND proxy



Because Host A's IPv6 address is on the same subnet as Host B's, Host A directly sends an NS message to obtain Host B's MAC address. However, Host B cannot receive the NS message because they are isolated at Layer 2.

To solve this problem, enable local ND proxy on VLAN-interface 2 of the switch A so that the switch A can forward messages between Host A and Host B.

Local ND proxy implements Layer 3 communication for two hosts in the following cases:

- The two hosts must connect to different isolated Layer 2 ports of a VLAN.
- If isolate-user-VLAN is used, the two hosts must belong to different secondary VLANs.

Configuration procedure

You can enable common ND proxy and local ND proxy in VLAN interface view.

To enable common ND proxy:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Enable common ND proxy.	proxy-nd enable	Disabled by default

To enable local ND proxy:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Enable local ND proxy.	local-proxy-nd enable	Optional. Disabled by default.

Configuring path MTU discovery

Configuring a static path MTU for a specific IPv6 address

You can configure a static path MTU for a specific destination IPv6 address. When a source host sends a packet through an interface, it compares the interface MTU with the static path MTU of the specified destination IPv6 address. If the packet size is larger than the smaller one of the two values, the host fragments the packet according to the smaller value.

To configure a static path MTU for a specific IPv6 address:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Configure a static path MTU for a specific IPv6 address.	ipv6 pathmtu <i>ipv6-address</i> [<i>value</i>]	Not configured by default

Configuring the aging time for dynamic path MTUs

After the path MTU from a source host to a destination host is dynamically determined (see "[IPv6 path MTU discovery](#)"), the source host sends subsequent packets to the destination host based on this MTU. After the aging time expires, the dynamic path MTU is removed and the source host re-determines a dynamic path MTU through the path MTU mechanism.

The aging time is invalid for a static path MTU.

To configure the aging time for dynamic path MTUs:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Configure the aging time for dynamic path MTUs.	ipv6 pathmtu age <i>age-time</i>	Optional. 10 minutes by default.

Configuring IPv6 TCP properties

You can configure the following IPv6 TCP properties:

- **synwait timer**—When a SYN packet is sent, the synwait timer is triggered. If no response packet is received before the synwait timer expires, the IPv6 TCP connection establishment fails.
- **finwait timer**—When the IPv6 TCP connection status is FIN_WAIT_2, the finwait timer is triggered. If no packet is received before the finwait timer expires, the IPv6 TCP connection is terminated. If a FIN packet is received, the IPv6 TCP connection status becomes TIME_WAIT. If non-FIN packets are received, the finwait timer is reset upon receipt of the last non-FIN packet and the connection is terminated after the finwait timer expires.
- **Size of the IPv6 TCP sending/receiving buffer**

To configure IPv6 TCP properties:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Set the synwait timer.	tcp ipv6 timer syn-timeout <i>wait-time</i>	Optional. 75 seconds by default.
3. Set the finwait timer.	tcp ipv6 timer fin-timeout <i>wait-time</i>	Optional. 675 seconds by default.
4. Set the size of the IPv6 TCP sending/receiving buffer.	tcp ipv6 window <i>size</i>	Optional. 8 KB by default.

Configuring ICMPv6 packet sending

Configuring the maximum ICMPv6 error packets sent in an interval

If too many ICMPv6 error packets are sent within a short time in a network, network congestion may occur. To avoid network congestion, you can control the maximum number of ICMPv6 error packets sent within a specific time by adopting the token bucket algorithm.

You can set the capacity of a token bucket to determine the number of tokens in the bucket. In addition, you can set the update interval of the token bucket, the interval for restoring the configured capacity. One token allows one ICMPv6 error packet to be sent. Each time an ICMPv6 error packet is sent, the number of tokens in a token bucket decreases by one. If the number of ICMPv6 error packets successively sent exceeds the capacity of the token bucket, the additional ICMPv6 error packets cannot be sent out until the capacity of the token bucket is restored.

To configure the capacity and update interval of the token bucket:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Configure the capacity and update interval of the token bucket.	ipv6 icmp-error { bucket <i>bucket-size ratelimit</i> interval } *	Optional. By default, the capacity of a token bucket is 10 and the update interval is 100 milliseconds. A maximum of 10 ICMPv6 error packets can be sent within 100 milliseconds. The update interval "0" indicates that the number of ICMPv6 error packets sent is not restricted.

Enabling replying to multicast echo requests

If hosts are configured to answer multicast echo requests, an attacker can use this mechanism to attack a host. For example, if Host A (an attacker) sends an echo request with the source being Host B to a multicast address, all hosts in the multicast group will send echo replies to Host B. To prevent such an attack, disable a device from answering multicast echo requests by default. In some application scenarios, however, you must enable the device to answer multicast echo requests.

To enable replying to multicast echo requests:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable replying to multicast echo requests.	ipv6 icmpv6 multicast-echo-reply enable	Not enabled by default

Enabling sending ICMPv6 time exceeded messages

A device sends out an ICMPv6 Time Exceeded message in the following situations:

- If a received IPv6 packet's destination IP address is not a local address and its hop limit is 1, the device sends an ICMPv6 Hop Limit Exceeded message to the source.
- Upon receiving the first fragment of an IPv6 datagram with the destination IP address being the local address, the device starts a timer. If the timer expires before all the fragments arrive, an ICMPv6 Fragment Reassembly Timeout message is sent to the source.

If large quantities of malicious packets are received, the performance of a device degrades greatly because it must send back ICMP Time Exceeded messages. You can disable sending ICMPv6 Time Exceeded messages.

To enable sending ICMPv6 time exceeded messages:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable sending ICMPv6 Time Exceeded messages.	ipv6 hoplimit-expires enable	Optional. Enabled by default.

Enabling sending ICMPv6 destination unreachable messages

If the device fails to forward a received IPv6 packet because of one of the following reasons, it drops the packet and sends a corresponding ICMPv6 Destination Unreachable error message to the source.

- If no route is available for forwarding the packet, the device sends a "no route to destination" ICMPv6 error message to the source.
- If the device fails to forward the packet because of an administrative prohibition (such as a firewall filter or an ACL), the device sends the source a "destination network administratively prohibited" ICMPv6 error message.
- If the device fails to deliver the packet because the destination is beyond the scope of the source IPv6 address (for example, the source IPv6 address of the packet is a link-local address whereas the destination IPv6 address of the packet is a global unicast address), the device sends the source a "beyond scope of source address" ICMPv6 error message.
- If the device fails to resolve the corresponding link layer address of the destination IPv6 address, the device sends the source an "address unreachable" ICMPv6 error message.
- If the packet with the destination being local and transport layer protocol being UDP and the packet's destination port number does not match the running process, the device sends the source a "port unreachable" ICMPv6 error message.

If an attacker sends abnormal traffic that causes the device to generate ICMPv6 destination unreachable messages, end users may be affected. To prevent such attacks, you can disable the device from sending ICMPv6 destination unreachable messages.

To enable sending ICMPv6 destination unreachable messages:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable sending ICMPv6 destination unreachable messages.	ipv6 unreachable enable	Disabled by default

Displaying and maintaining IPv6 basics configuration

Task	Command	Remarks
Display the IPv6 FIB entries.	display ipv6 fib [acl6 <i>acl6-number</i> ipv6-prefix <i>ipv6-prefix-name</i>] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display the IPv6 FIB entry of a specific destination IPv6 address.	display ipv6 fib <i>ipv6-address</i> [<i>prefix-length</i>] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display IPv6 information about the interface.	display ipv6 interface [<i>interface-type</i> [<i>interface-number</i>]] [brief] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display neighbor information .	display ipv6 neighbors { { <i>ipv6-address</i> all dynamic static } [<i>slot slot-number</i>] interface <i>interface-type interface-number</i> vlan <i>vlan-id</i> } [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display the total number of neighbor entries satisfying the specified conditions .	display ipv6 neighbors { { all dynamic static } [<i>slot slot-number</i>] interface <i>interface-type interface-number</i> vlan <i>vlan-id</i> } count [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display IPv6 path MTU information.	display ipv6 pathmtu { <i>ipv6-address</i> all dynamic static } [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display socket information .	display ipv6 socket [socket-type <i>socket-type</i>] [<i>task-id socket-id</i>] [<i>slot slot-number</i>] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display the statistics of IPv6 packets and ICMPv6 packets .	display ipv6 statistics [<i>slot slot-number</i>] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display the IPv6 TCP connection statistics.	display tcp ipv6 statistics [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display the IPv6 TCP connection status information.	display tcp ipv6 status [{ begin exclude include } <i>regular-expression</i>]	Available in any view

Task	Command	Remarks
Display the IPv6 UDP connection statistics.	display udp ipv6 statistics [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display ND snooping entries.	display ipv6 nd snooping [<i>ipv6-address</i> vlan <i>vlan-id</i>] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Clear IPv6 neighbor information .	reset ipv6 neighbors { all dynamic interface <i>interface-type interface-number</i> slot <i>slot-number</i> static }	Available in user view
Clear the path MTU values.	reset ipv6 pathmtu { all static dynamic }	Available in user view
Clear the statistics of IPv6 and ICMPv6 packets .	reset ipv6 statistics [slot <i>slot-number</i>]	Available in user view
Clear all IPv6 TCP connection statistics.	reset tcp ipv6 statistics	Available in user view
Clear the statistics of all IPv6 UDP packets.	reset udp ipv6 statistics	Available in user view
Clear ND snooping entries.	reset ipv6 nd snooping [<i>ipv6-address</i> vlan <i>vlan-id</i>]	Available in user view

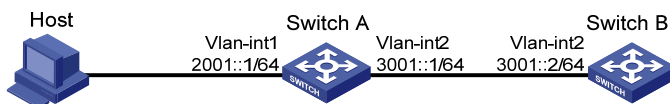
IPv6 basics configuration example

Network requirements

As shown in [Figure 57](#), a host, Switch A and Switch B are connected through Ethernet ports. Add the Ethernet ports into corresponding VLANs, configure IPv6 addresses for the VLAN interfaces and verify that they are connected.

- The global unicast addresses of VLAN-interface 1 and VLAN-interface 2 on Switch A are 2001::1/64 and 3001::1/64, respectively.
- The global unicast address of VLAN-interface 2 on Switch B is 3001::2/64, and a route to Host is available.
- IPv6 is enabled for the host to automatically obtain an IPv6 address through IPv6 ND, and a route to Switch B is available.

Figure 57 Network diagram



The VLAN interfaces have been created on the switch.

Configuration procedure

1. Configure Switch A:


```
# Enable IPv6.
<SwitchA> system-view
```

```
[SwitchA] ipv6
# Specify a global unicast address for VLAN-interface 2.
[SwitchA] interface vlan-interface 2
[SwitchA-Vlan-interface2] ipv6 address 3001::1/64
[SwitchA-Vlan-interface2] quit
# Specify a global unicast address for VLAN-interface 1, and allow it to advertise RA messages (no
interface advertises RA messages by default).
[SwitchA] interface vlan-interface 1
[SwitchA-Vlan-interface1] ipv6 address 2001::1/64
[SwitchA-Vlan-interface1] undo ipv6 nd ra halt
[SwitchA-Vlan-interface1] quit
```

2. Configure Switch B:

```
# Enable IPv6.
<SwitchB> system-view
[SwitchB] ipv6
# Configure a global unicast address for VLAN-interface 2.
[SwitchB] interface vlan-interface 2
[SwitchB-Vlan-interface2] ipv6 address 3001::2/64
[SwitchB-Vlan-interface2] quit
# Configure an IPv6 static route with destination IP address 2001::/64 and next hop address
3001::1.
[SwitchB] ipv6 route-static 2001:: 64 3001::1
```

3. Configure the host:

```
# Enable IPv6 for Host to automatically obtain an IPv6 address through IPv6 ND.
# On Switch A, use the ping ipv6 command to ping Switch B for the connectivity.
```

```
[SwitchA] ping ipv6 3001::1
  PING 3001::1 : 56 data bytes, press CTRL_C to break
    Reply from 3001::1
    bytes=56 Sequence=0 hop limit=64 time = 3 ms
    Reply from 3001::1
    bytes=56 Sequence=1 hop limit=64 time = 2 ms
    Reply from 3001::1
    bytes=56 Sequence=2 hop limit=64 time = 2 ms
    Reply from 3001::1
    bytes=56 Sequence=3 hop limit=64 time = 3 ms
    Reply from 3001::1
    bytes=56 Sequence=4 hop limit=64 time = 9 ms

--- 3001::1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 2/3/9 ms
```

```
# Display neighbor information about GigabitEthernet 1/0/2 on Switch A.
```

```
[SwitchA] display ipv6 neighbors interface GigabitEthernet 1/0/2
Type: S-Static D-Dynamic
```

IPv6 Address	Link-layer	VID	Interface	State	T	Age
FE80::215:E9FF:FEA6:7D14	0015-e9a6-7d14	1	GE1/0/2	STALE	D	1238
2001::15B:E0EA:3524:E791	0015-e9a6-7d14	1	GE1/0/2	STALE	D	1248

The output shows that the IPv6 global unicast address that the host obtained is 2001::15B:E0EA:3524:E791.

Verifying the configuration

Display the IPv6 interface settings on Switch A. All of the IPv6 global unicast addresses configured on the interface are displayed.

```
[SwitchA] display ipv6 interface vlan-interface 2
Vlan-interface2 current state :UP
Line protocol current state :UP
IPv6 is enabled, link-local address is FE80::20F:E2FF:FE00:2
Global unicast address(es):
  3001::1, subnet is 3001::/64
Joined group address(es):
  FF02::1:FF00:0
  FF02::1:FF00:1
  FF02::1:FF00:2
  FF02::2
  FF02::1
MTU is 1500 bytes
ND DAD is enabled, number of DAD attempts: 1
ND reachable time is 30000 milliseconds
ND retransmit interval is 1000 milliseconds
Hosts use stateless autoconfig for addresses
IPv6 Packet statistics:
InReceives:                25829
InTooShorts:                0
InTruncatedPkts:           0
InHopLimitExceeds:         0
InBadHeaders:               0
InBadOptions:               0
ReasmReqds:                 0
ReasmOKs:                   0
InFragDrops:                0
InFragTimeouts:             0
OutFragFails:                0
InUnknownProtos:            0
InDelivers:                  47
OutRequests:                 89
OutForwDatagrams:           48
InNoRoutes:                  0
InTooBigErrors:              0
OutFragOKs:                  0
OutFragCreates:             0
InMcastPkts:                 6
```

```
InMcastNotMembers:      25747
OutMcastPkts:           48
InAddrErrors:           0
InDiscards:             0
OutDiscards:            0
```

[SwitchA] display ipv6 interface vlan-interface 1

Vlan-interface1 current state :UP

Line protocol current state :UP

IPv6 is enabled, link-local address is FE80::20F:E2FF:FE00:1C0

Global unicast address(es):

2001::1, subnet is 2001::/64

Joined group address(es):

FF02::1:FF00:0

FF02::1:FF00:1

FF02::1:FF00:1C0

FF02::2

FF02::1

MTU is 1500 bytes

ND DAD is enabled, number of DAD attempts: 1

ND reachable time is 30000 milliseconds

ND retransmit interval is 1000 milliseconds

ND advertised reachable time is 0 milliseconds

ND advertised retransmit interval is 0 milliseconds

ND router advertisements are sent every 600 seconds

ND router advertisements live for 1800 seconds

Hosts use stateless autoconfig for addresses

IPv6 Packet statistics:

```
InReceives:             272
InTooShorts:            0
InTruncatedPkts:       0
InHopLimitExceeds:     0
InBadHeaders:          0
InBadOptions:          0
ReasmReqds:            0
ReasmOKs:              0
InFragDrops:           0
InFragTimeouts:       0
OutFragFails:          0
InUnknownProtos:      0
InDelivers:            159
OutRequests:           1012
OutForwDatagrams:      35
InNoRoutes:            0
InTooBigErrors:        0
OutFragOKs:            0
OutFragCreates:        0
InMcastPkts:           79
```

```
InMcastNotMembers:      65
OutMcastPkts:           938
InAddrErrors:           0
InDiscards:             0
OutDiscards:            0
```

Display the IPv6 interface settings on Switch B. All the IPv6 global unicast addresses configured on the interface are displayed.

```
[SwitchB] display ipv6 interface vlan-interface 2
Vlan-interface2 current state :UP
Line protocol current state :UP
IPv6 is enabled, link-local address is FE80::20F:E2FF:FE00:1234
Global unicast address(es):
  3001::2, subnet is 3001::/64
Joined group address(es):
  FF02::1:FF00:0
  FF02::1:FF00:2
  FF02::1:FF00:1234
  FF02::2
  FF02::1
MTU is 1500 bytes
ND DAD is enabled, number of DAD attempts: 1
ND reachable time is 30000 milliseconds
ND retransmit interval is 1000 milliseconds
Hosts use stateless autoconfig for addresses
IPv6 Packet statistics:
InReceives:              117
InTooShorts:             0
InTruncatedPkts:         0
InHopLimitExceeds:       0
InBadHeaders:            0
InBadOptions:            0
ReasmReqds:              0
ReasmOKs:                 0
InFragDrops:             0
InFragTimeouts:          0
OutFragFails:            0
InUnknownProtos:         0
InDelivers:              117
OutRequests:             83
OutForwDatagrams:        0
InNoRoutes:              0
InTooBigErrors:          0
OutFragOKs:              0
OutFragCreates:          0
InMcastPkts:            28
InMcastNotMembers:       0
OutMcastPkts:            7
```

```
InAddrErrors:          0
InDiscards:           0
OutDiscards:          0
```

Ping Switch A and Switch B on the host, and ping Switch A and the host on Switch B to verify that they are connected.

△ IMPORTANT:

When you ping a link-local address, you should use the `-i` parameter to specify an interface for the link-local address.

```
[SwitchB] ping ipv6 -c 1 3001::1
PING 3001::1 : 56 data bytes, press CTRL_C to break
  Reply from 3001::1
    bytes=56 Sequence=1 hop limit=64 time = 2 ms

--- 3001::1 ping statistics ---
  1 packet(s) transmitted
  1 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 2/2/2 ms
[SwitchB-Vlan-interface2] ping ipv6 -c 1 2001::15B:E0EA:3524:E791
PING 2001::15B:E0EA:3524:E791 : 56 data bytes, press CTRL_C to break
  Reply from 2001::15B:E0EA:3524:E791
    bytes=56 Sequence=1 hop limit=63 time = 3 ms

--- 2001::15B:E0EA:3524:E791 ping statistics ---
  1 packet(s) transmitted
  1 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 3/3/3 ms
```

The output shows that Switch B can ping Switch A and the host.

Troubleshooting IPv6 basics configuration

Symptom

The peer IPv6 address cannot be pinged.

Solution

1. Use the **display current-configuration** command in any view or the **display this** command in system view to verify that IPv6 is enabled.
2. Use the **display ipv6 interface** command in any view to verify that the IPv6 address of the interface is correct and the interface is up.
3. Use the **debugging ipv6 packet** command in user view to enable the debugging for IPv6 packets to help locate the cause.

DHCPv6 overview

Introduction to DHCPv6

The Dynamic Host Configuration Protocol for IPv6 (DHCPv6) was designed based on IPv6 addressing scheme and is used for assigning IPv6 prefixes, IPv6 addresses and other configuration parameters to hosts.

Compared with other IPv6 address allocation methods (such as manual configuration and stateless address autoconfiguration), DHCPv6 can:

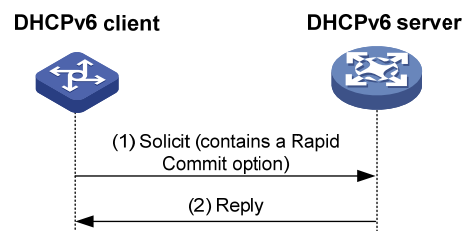
- Record addresses assigned to hosts and assign specific addresses to hosts, thus facilitating network management.
- Assign prefixes to devices, facilitating automatic configuration and management of the entire network.
- Assign other configuration parameters, such as DNS server addresses and domain names.

DHCPv6 address/prefix assignment

A process of DHCPv6 address/prefix assignment involves two or four messages. The following describe the detailed processes.

Rapid assignment involving two messages

Figure 58 Rapid assignment involving two messages



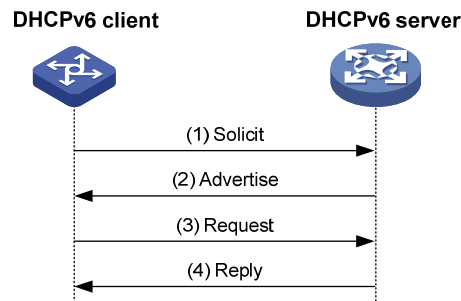
As shown in [Figure 58](#), the rapid assignment involving two messages operates in the following steps.

1. The DHCPv6 client sends out a Solicit message that contains a Rapid Commit option, requesting that rapid assignment of address/prefix and other configuration parameters should be preferred.
2. If the DHCPv6 server supports rapid assignment, it responds with a Reply message containing the assigned IPv6 address/prefix and other configuration parameters. If the DHCPv6 server does not support rapid assignment, [Assignment involving four messages](#) is implemented.

Assignment involving four messages

[Figure 59](#) shows the process of IPv6 address/prefix assignment involving four messages.

Figure 59 Assignment involving four messages



The assignment involving four messages operates in the following steps:

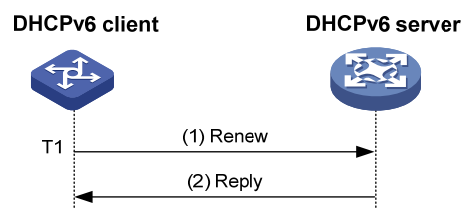
1. The DHCPv6 client sends out a Solicit message, requesting an IPv6 address/prefix and other configuration parameters.
2. If the Solicit message does not contain a Rapid Commit option, or if the DHCPv6 server does not support rapid assignment even though the Solicit message contains a Rapid Commit option, the DHCPv6 server responds with an Advertise message, informing the DHCPv6 client of the assignable address/prefix and other configuration parameters.
3. The DHCPv6 client may receive multiple Advertise messages offered by different DHCPv6 servers. It then selects an offer according to the receiving sequence and server priority, and sends a Request message to the selected server for the confirmation of assignment.
4. The DHCPv6 server sends a Reply message to the client, confirming that the address/prefix and other configuration parameters are assigned to the client.

Address/prefix lease renewal

The IPv6 address/prefix assigned by the DHCPv6 server has a lease time, which depends on the valid lifetime. When the valid lifetime of the IPv6 address/prefix expires, the DHCPv6 client cannot use the IPv6 address/prefix any longer. To continue using the IPv6 address/prefix, the DHCPv6 client has to renew the lease time.

As shown in [Figure 60](#), at T1, the DHCPv6 client unicasts a Renew message to the DHCPv6 server that assigned the IPv6 address/prefix to the DHCPv6 client. The recommended value of T1 is half the preferred lifetime. Then the DHCPv6 server responds with a Reply message, informing the client about whether or not the lease is renewed.

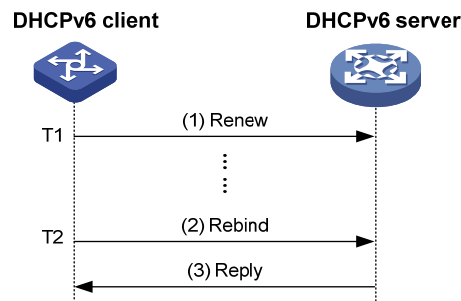
Figure 60 Using the Renew message for address/prefix lease renewal



As shown in [Figure 61](#), if the DHCPv6 client receives no response from the DHCPv6 server after sending out a Renew message at T1, it multicasts a Rebind message to all DHCPv6 servers at T2 (that is, when 80% preferred lifetime expires). Then the DHCPv6 server responds with a Reply message, informing the client about whether or not the lease is renewed.

If the DHCPv6 client receives no response from the DHCPv6 servers, the client stops using the address/prefix when the valid lifetime expires. For more information about the valid lifetime and the preferred lifetime, see "Configuring IPv6 basics."

Figure 61 Using the Rebind message for address/prefix lease renewal



Configuring stateless DHCPv6

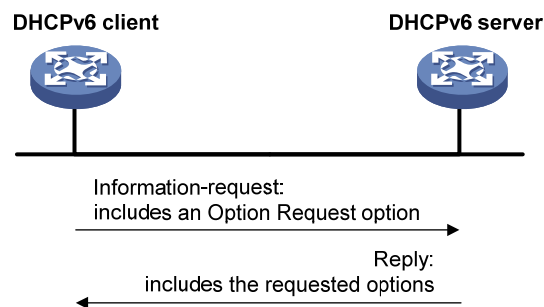
After obtaining an IPv6 address/prefix, a device can use stateless DHCPv6 to obtain other configuration parameters from a DHCPv6 server. This application is called stateless DHCPv6 configuration.

With an IPv6 address obtained through stateless address autoconfiguration, a device automatically enables the stateless DHCPv6 function after it receives an RA message with the managed address configuration flag (M flag) set to 0 and with the other stateful configuration flag (O flag) set to 1.

Stateless address autoconfiguration means that a node automatically generates an IPv6 address based on the information obtained through router/prefix discovery. For more information, see "Configuring IPv6 basics."

Operation

Figure 62 Operation of stateless DHCPv6



As shown in [Figure 62](#), stateless DHCPv6 operates in the following steps:

1. The DHCPv6 client multicasts an Information-request message to the multicast address of all DHCPv6 servers and DHCPv6 relay agents. The Information-request message contains an Option Request option, specifying the configuration parameters that the client requests from the DHCPv6 server.
2. After receiving the Information-request message, the DHCPv6 server returns the client a Reply message containing the requested configuration parameters.
3. The client checks the Reply message. If the obtained configuration parameters match those requested in the Information-request message, the client performs network configuration with the

parameters. If not, the client ignores the configuration parameters. If multiple replies are received, the first received reply will be used.

Protocols and standards

- RFC 3736, *Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6*
- RFC 3315, *Dynamic Host Configuration Protocol for IPv6 (DHCPv6)*
- RFC 2462, *IPv6 Stateless Address Autoconfiguration*
- RFC 3633, *IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6*

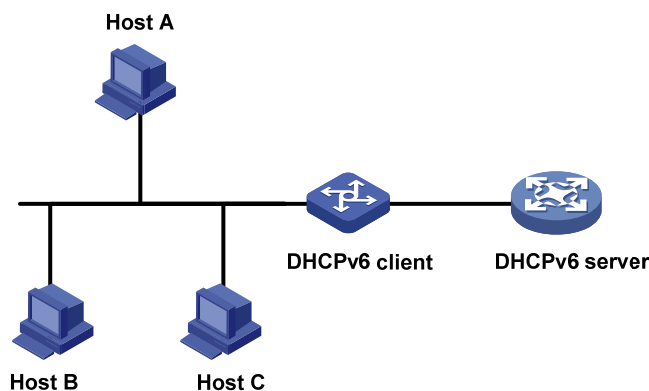
Configuring DHCPv6 server

Overview

As shown in [Figure 63](#), the DHCPv6 server assigns the DHCPv6 client an IPv6 prefix to facilitate IPv6 address management and network configuration. After obtaining the IPv6 prefix, the DHCPv6 client sends an RA message containing the prefix information to the subnet where it resides, so that hosts on the subnet can automatically configure their IPv6 addresses by using the prefix.

A device serving as a DHCPv6 server assigns DHCPv6 clients IPv6 prefixes, but not IPv6 addresses, and supports DHCPv6 stateless configuration to assign other configuration parameters.

Figure 63 Typical DHCPv6 server application



Concepts

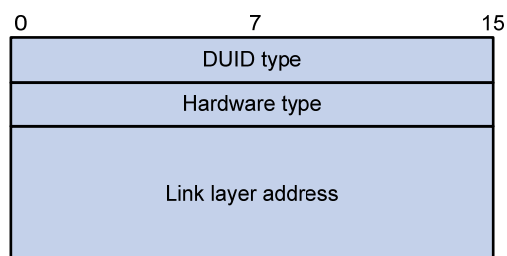
DHCPv6 multicast address

The multicast address FF05::1:3 identifies all DHCPv6 servers on the site-local network. The multicast address FF02::1:2 identifies all DHCPv6 servers and relay agents on the link-local link.

DUID

A DHCP unique identifier (DUID) uniquely identifies a DHCPv6 device (DHCPv6 client, server, or relay agent).

Figure 64 DUID-LL format



A DUID based on link-layer address (DUID-LL) defined in RFC 3315 is used to identify a DHCPv6 device. Figure 64 shows the DUID-LL format, where:

- **DUID type**—The device supports DUID-LL as the DUID type with the value of 0x0003.
- **Hardware type**—The device supports Ethernet as the hardware type with the value of 0x0001.
- **Link layer address**—Its value is the bridge MAC address of the device.

IA

Identified by an IAID, an Identity Association (IA) provides a construct through which the obtained addresses, prefixes, and other configuration parameters assigned from a server to a client are managed. A client can maintain multiple IAs, each of which is configured on an interface to manage the addresses, prefixes, and other configuration parameters obtained by that interface.

IAID

An IAID uniquely identifies an IA. It is chosen by the client and must be unique among the IAIDs on the client.

PD

The Prefix Delegation (PD) is the lease record created by the DHCPv6 server for each assigned prefix. The PD contains information such as the IPv6 prefix, client DUID, IAID, valid lifetime, preferred lifetime, lease expiration time, and the IPv6 address of the requesting client.

Prefix selection process

Upon receiving a request, the DHCPv6 server selects the prefix and other configuration parameters from the address pool that is applied to the interface receiving the request. An address pool may contain the static prefixes configured for specific clients, or have a prefix pool referenced for dynamic assignment from the specific prefix range.

A DHCPv6 server selects a prefix from the address pool according to the following sequence:

1. The desired static prefix with the DUID and IAID matching those of the client
2. The static prefix with the DUID and IAID matching those of the client
3. The desired static prefix with the DUID matching the client's DUID and with no client IAID specified
4. The static prefix with the DUID matching the client's DUID and with no client IAID specified
5. The desired idle prefix in the prefix pool
6. An idle prefix in the prefix pool

DHCPv6 server configuration task list

Before you configure the DHCPv6 server, enable IPv6 by using the **ipv6** command.

Task	Remarks
Enabling the DHCPv6 server	Required
Creating a prefix pool	Required
Configuring a DHCPv6 address pool	Required
Applying the address pool to an interface	Required
Setting the DSCP value for DHCPv6 packets	Optional

Enabling the DHCPv6 server

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable the DHCPv6 server function.	ipv6 dhcp server enable	Disabled by default

Creating a prefix pool

A prefix pool specifies a range of prefixes.

To create a prefix pool:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create a prefix pool.	ipv6 dhcp prefix-pool <i>prefix-pool-number</i> prefix <i>prefix/prefix-len</i> assign-len <i>assign-len</i>	Not configured by default

Configuring a DHCPv6 address pool

You can configure prefixes and other configuration parameters, such as the DNS server address, domain name, SIP server address, domain name of the SIP server, and address family translation router (AFTR) in a DHCPv6 address pool, for the DHCPv6 server to assign them to DHCPv6 clients.

Configuration restrictions and guidelines

- Only one prefix pool can be referenced by an address pool.
- A non-existing prefix pool can be referenced by an address pool. However, no prefix is available in the prefix pool for dynamic prefix assignment until the prefix pool is created.
- You cannot modify the prefix pool referenced by an address pool, or the preferred lifetime or valid lifetime by using the **prefix-pool** command. You must remove the configuration before you can have another prefix pool referenced by the address pool, or modify the preferred lifetime and valid lifetime.
- You can configure up to eight DNS server addresses, one domain name, eight SIP server addresses, and eight SIP server domain names in an address pool.

Configuration procedure

To configure a DHCPv6 address pool:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create a DHCPv6 address pool and enter DHCPv6 address pool view.	ipv6 dhcp pool <i>pool-number</i>	Not configured by default.

Step	Command	Remarks
3. Configure a DHCPv6 address pool.	<ul style="list-style-type: none"> Configure a static prefix: static-bind prefix <i>prefix/prefix-len</i> duid <i>duid</i> [iaid <i>iaid</i>] [preferred-lifetime <i>preferred-lifetime</i> valid-lifetime <i>valid-lifetime</i>] Apply a prefix pool to the address pool: prefix-pool <i>prefix-pool-number</i> [preferred-lifetime <i>preferred-lifetime</i> valid-lifetime <i>valid-lifetime</i>] 	Use either command. No prefix is specified by default.
4. Configure a DNS server address.	dns-server <i>ipv6-address</i>	Optional. Not configured by default.
5. Configure a domain name.	domain-name <i>domain-name</i>	Optional. Not configured by default.
6. Configure the IPv6 address or domain name of a SIP server.	sip-server { address <i>ipv6-address</i> domain-name <i>domain-name</i> }	Optional. Not configured by default.
7. Specify the AFTR address.	ds-lite address <i>ipv6-address</i>	Optional. Not specified by default.

Applying the address pool to an interface

After an address pool is applied to an interface, a prefix and other configuration parameters can be selected from the address pool and assigned to the DHCPv6 client requesting through the interface.

Follow these guidelines when you apply an address pool to an interface:

- An interface cannot serve as a DHCPv6 server and DHCPv6 relay agent at the same time.
- It is not recommended that you enable DHCPv6 server and DHCPv6 client on the same interface.
- Only one address pool can be applied to an interface.
- A non-existing address pool can be applied to an interface. However, the server cannot assign any prefix or other configuration information from the address pool until the address pool is created.
- You cannot modify the address pool applied to an interface or parameters such as the server priority by using the **ipv6 dhcp server apply pool** command. You must remove the applied address pool before you can apply another address pool to the interface or modify parameters such as the server priority.

To apply an address pool to an interface:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type interface-number</i>	N/A
3. Apply the DHCPv6 address pool to the interface.	ipv6 dhcp server apply pool <i>pool-number</i> [allow-hint preference <i>preference-value</i> rapid-commit] *	Not configured by default

Setting the DSCP value for DHCPv6 packets

An IPv6 packet header contains an 8-bit Traffic class field. This field identifies the service type of IPv6 packets. As defined in RFC 2474, the first six bits set the Differentiated Services Code Point (DSCP) value, and the last two bits are reserved. Network devices use the DSCP value as a reference to determine the packet priority for transmission.

To set the DSCP value for DHCPv6 packets:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Set the DSCP value for DHCPv6 packets sent by the DHCPv6 server.	ipv6 dhcp dscp <i>dscp-value</i>	Optional. By default, the DSCP value in DHCPv6 packets is 56.

Displaying and maintaining the DHCPv6 server

Task	Command	Remarks
Display the DUID of the local device.	display ipv6 dhcp duid [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display DHCPv6 address pool information.	display ipv6 dhcp pool [<i>pool-number</i>] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display prefix pool information.	display ipv6 dhcp prefix-pool [<i>prefix-pool-number</i>] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display DHCPv6 server configuration information.	display ipv6 dhcp server [<i>interface interface-type interface-number</i>] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display PD information.	display ipv6 dhcp server pd-in-use { all pool <i>pool-number</i> prefix <i>prefix/prefix-len</i> prefix-pool <i>prefix-pool-number</i> } [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display packet statistics on the DHCPv6 server.	display ipv6 dhcp server statistics [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Clear PD information on the DHCPv6 server.	reset ipv6 dhcp server pd-in-use { all pool <i>pool-number</i> prefix <i>prefix/prefix-len</i> }	Available in user view
Clear packets statistics on the DHCPv6 server.	reset ipv6 dhcp server statistics	Available in user view

DHCPv6 server configuration example

Network requirements

As shown in [Figure 65](#), the switch serves as a DHCPv6 server, and assigns the IPv6 prefix, DNS server address, domain name, SIP server address, and SIP server domain name to the DHCPv6 clients. The IPv6 address of the switch is 1::1/64.

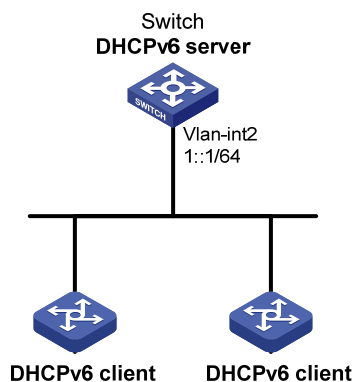
The switch assigns prefix 2001:0410:0201::/48 to the client whose DUID is 00030001CA0006A40000, and assigns prefixes ranging from 2001:0410::/48 to 2001:0410:FFFF::/48 (excluding 2001:0410:0201::/48) to other clients. The DNS server address is 2::2:3. The DHCPv6 clients reside in domain aaa.com. The SIP server address is 2:2::4, and the domain name of the SIP server is bbb.com.

Configuration considerations

To configure the DHCPv6 server:

1. Enable IPv6 and DHCPv6 server.
2. Create a prefix pool containing prefix 2001:0410::/32 with the length of the assigned prefix being 48, so that the server assigns clients the prefixes ranging 2001:0410::/48 to 2001:0410:FFFF::/48.
3. Create an address pool. Configure a static prefix in the address pool and have the prefix pool referenced by the address pool. Configure other configuration parameters.
4. Apply the address pool to the interface through which the server is connected to the clients.

Figure 65 Network diagram



Configuration procedure

```
# Enable IPv6 and DHCPv6 server.
<Switch> system-view
[Switch] ipv6
[Switch] ipv6 dhcp server enable

# Configure the IPv6 address of VLAN-interface 2.
[Switch] interface vlan-interface 2
[Switch-Vlan-interface2] ipv6 address 1::1/64
```

```

[Switch-Vlan-interface2] quit
# Create and configure prefix pool 1.
[Switch] ipv6 dhcp prefix-pool 1 prefix 2001:0410::/32 assign-len 48
# Create address pool 1.
[Switch] ipv6 dhcp pool 1
# Apply prefix pool 1 to address pool 1, and set the preferred lifetime to one day, the valid lifetime to three days.
[Switch-ipv6-dhcp-pool-1] prefix-pool 1 preferred-lifetime 86400 valid-lifetime 259200
# Configure static prefix 2001:0410:0201::/48 in address pool 1, and set the client DUID as 00030001CA0006A40000, the preferred lifetime to one day, and the valid lifetime to three days.
[Switch-ipv6-dhcp-pool-1] static-bind prefix 2001:0410:0201::/48 duid 00030001CA0006A40000 preferred-lifetime 86400 valid-lifetime 259200
# Configure the DNS server address as 2:2::3.
[Switch-ipv6-dhcp-pool-1] dns-server 2:2::3
# Configure the domain name as aaa.com.
[Switch-ipv6-dhcp-pool-1] domain-name aaa.com
# Configure the SIP server address as 2:2::4, and the domain name of the SIP server as bbb.com.
[Switch-ipv6-dhcp-pool-1] sip-server address 2:2::4
[Switch-ipv6-dhcp-pool-1] sip-server domain-name bbb.com
[Switch-ipv6-dhcp-pool-1] quit
# Apply address pool 1 to VLAN-interface 2, configure the address pool to support the desired prefix assignment and rapid prefix assignment, and set the precedence to the highest.
[Switch] interface vlan-interface 2
[Switch-Vlan-interface2] ipv6 dhcp server apply pool 1 allow-hint preference 255
rapid-commit

```

Verifying the configuration

```

# Display DHCPv6 server configuration information on VLAN-interface 2.
[Switch-Vlan-interface2] display ipv6 dhcp server interface vlan-interface 2
Using pool: 1
Preference value: 255
Allow-hint: Enabled
Rapid-commit: Enabled
# Display information about address pool 1.
[Switch-Vlan-interface2] display ipv6 dhcp pool 1
DHCPv6 pool: 1
  Static bindings:
    DUID: 00030001CA0006A40000
    IAID: A1A1A1A1
    Prefix: 2001:410:201::/48
      preferred lifetime 86400, valid lifetime 2592000
  Prefix pool: 1
    preferred lifetime 86400, valid lifetime 2592000
  DNS server address:
    2:2::3

```

```
Domain name: aaa.com
SIP server address:
  2:2::4
SIP server domain name:
  bbb.com
```

Display information about prefix pool 1.

```
[Switch-Vlan-interface2] display ipv6 dhcp prefix-pool 1
Prefix: 2001:410::/32
Assigned length: 48
Total prefix number: 65536
Available: 65535
In-use: 0
Static: 1
```

After the client whose DUID is 00030001CA0006A40000 obtains an IPv6 prefix, display the PD information on the DHCPv6 server.

```
[Switch-Vlan-interface2] display ipv6 dhcp server pd-in-use all
Total number = 1
Prefix                                Type      Pool Lease-expiration
2001:410:201::/48                     Static(C) 1      Jul 10 2009 19:45:01
```

After the other client obtains an IPv6 prefix, display the PD information on the DHCPv6 server.

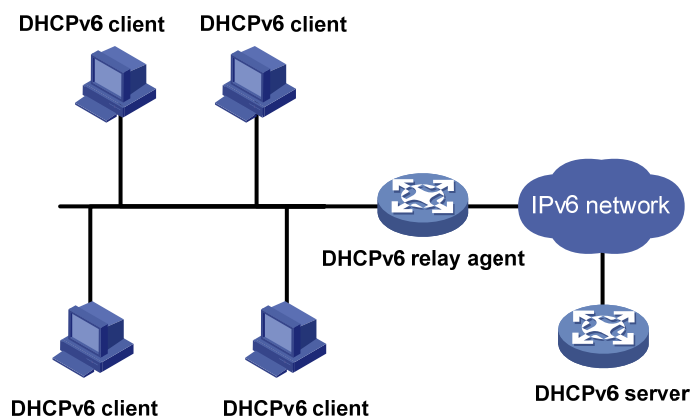
```
[Switch-Vlan-interface2] display ipv6 dhcp server pd-in-use all
Total number = 2
Prefix                                Type      Pool Lease-expiration
2001:410:201::/48                     Static(C) 1      Jul 10 2009 19:45:01
2001:410::/48                          Auto(C)  1      Jul 10 2009 20:44:05
```

Configuring DHCPv6 relay agent

Overview

A DHCPv6 client usually uses a multicast address to contact the DHCPv6 server on the local link to obtain an IPv6 address and other configuration parameters. As shown in [Figure 66](#), if the DHCPv6 server resides on another subnet, the DHCPv6 client can contact the server via a DHCPv6 relay agent, so you do not need to deploy a DHCPv6 server on each subnet.

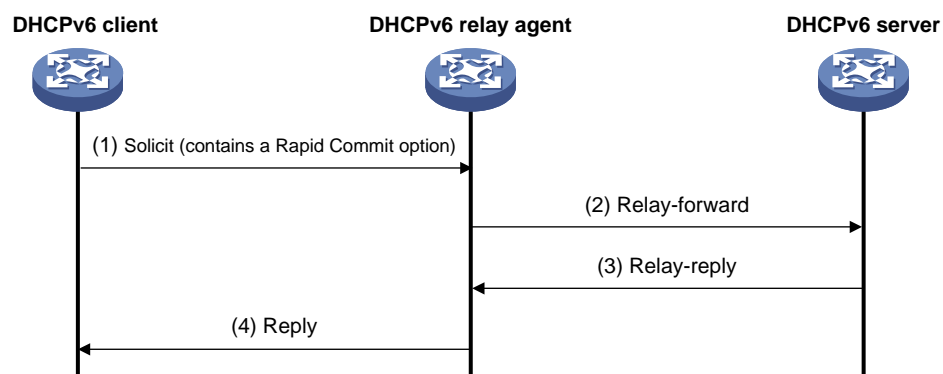
Figure 66 Typical DHCPv6 relay agent application



DHCPv6 relay agent operation

[Figure 67](#) shows how the DHCPv6 client obtains an IPv6 address and other network configuration parameters from the DHCPv6 server through the DHCPv6 relay agent, using the process of rapid assignment involving two messages.

Figure 67 Operating process of a DHCPv6 relay agent



The operation process is as follows:

1. The DHCPv6 client sends a Solicit message containing the Rapid Commit option to the multicast address FF02::1:2 of all the DHCPv6 servers and relay agents.

2. After receiving the Solicit message, the DHCPv6 relay agent encapsulates the message into the Relay Message option of a Relay-forward message, and sends the message to the DHCPv6 server.
3. After obtaining the Solicit message from the Relay-forward message, the DHCPv6 server selects an IPv6 address and other required parameters, and adds them to the reply which is encapsulated within the Relay Message option of a Relay-reply message. The DHCPv6 server then sends the Relay-reply message to the DHCPv6 relay agent.
4. The DHCPv6 relay agent obtains the reply from the Relay-reply message and sends the reply to the DHCPv6 client.

The DHCPv6 client uses the IPv6 address and other network parameters assigned by the DHCPv6 server to perform network configuration.

Configuring the DHCPv6 relay agent

Upon receiving a Solicit message from a DHCPv6 client, the interface that operates as a DHCPv6 relay agent encapsulates the request into a Relay-forward message and forwards the message to the specified DHCPv6 server, which then assigns an IPv6 address and other configuration parameters to the DHCPv6 client.

Configuration guidelines

Follow these guidelines when you configure the DHCPv6 relay agent:

- Before you configure the DHCPv6 relay agent, enable IPv6 by using the **ipv6** command in system view.
- Executing the **ipv6 dhcp relay server-address** command repeatedly can specify multiple DHCPv6 servers. Up to eight DHCPv6 servers can be specified for an interface. After receiving requests from DHCPv6 clients, the DHCPv6 relay agent forwards the requests to all the specified DHCPv6 servers.
- If the DHCPv6 server address is a link-local address or link-scoped multicast address on the local link, you must specify an outgoing interface using the **interface** keyword in the **ipv6 dhcp relay server-address** command. Otherwise, DHCPv6 packets may fail to be forwarded to the DHCPv6 server.
- After you remove all the specified DHCPv6 servers from an interface with the **undo ipv6 dhcp relay server-address** command, DHCPv6 relay agent is disabled on the interface.
- An interface cannot serve as a DHCPv6 relay agent and DHCPv6 server at the same time.
- H3C does not recommend enabling the DHCPv6 relay agent and DHCPv6 client on the same interface

Configuration procedure

To configure the DHCPv6 relay agent:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A

Step	Command	Remarks
3. Enable DHCPv6 relay agent on the interface and specify a DHCPv6 server.	ipv6 dhcp relay server-address <i>ipv6-address [interface interface-type interface-number]</i>	By default, DHCPv6 relay agent is disabled and no DHCPv6 server is specified on the interface.

Setting the DSCP value for DHCPv6 packets

An IPv6 packet header contains an 8-bit Traffic class field. This field identifies the service type of IPv6 packets. As defined in RFC 2474, the first six bits set the Differentiated Services Code Point (DSCP) value, and the last two bits are reserved. Network devices use the DSCP value as a reference to determine the packet priority for transmission.

To set the DSCP value for DHCPv6 packets:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Set the DSCP value for DHCPv6 packets sent by the DHCPv6 relay agent.	ipv6 dhcp dscp <i>dscp-value</i>	Optional. By default, the DSCP value in DHCPv6 packets is 56.

Displaying and maintaining the DHCPv6 relay agent

Task	Command	Remarks
Display the DUID of the local device.	display ipv6 dhcp duid [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display DHCPv6 server addresses specified on the DHCPv6 relay agent.	display ipv6 dhcp relay server-address { all interface <i>interface-type interface-number</i> } [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display packet statistics on the DHCPv6 relay agent.	display ipv6 dhcp relay statistics [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Clear packets statistics on the DHCPv6 relay agent.	reset ipv6 dhcp relay statistics	Available in user view

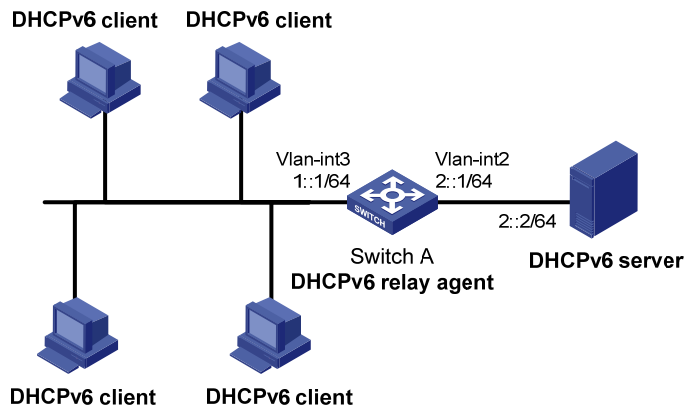
DHCPv6 relay agent configuration example

Network requirements

As shown in [Figure 68](#), the network address prefix of DHCPv6 clients is 1::/64, and the IPv6 address of the DHCPv6 server is 2::2/64. The DHCPv6 client and server need to communicate via a DHCPv6 relay agent (Switch A).

Switch A acts as the gateway of network 1::/64. It sends RA messages to notify the hosts to obtain IPv6 addresses and other configuration parameters through DHCPv6.

Figure 68 Network diagram



Configuration procedure

1. Configure Switch A as a DHCPv6 relay agent:

Enable the IPv6 packet forwarding function.

```
<SwitchA> system-view
[SwitchA] ipv6
```

Configure the IPv6 addresses of VLAN-interface 2 and VLAN-interface 3, respectively.

```
[SwitchA] interface vlan-interface 2
[SwitchA-Vlan-interface2] ipv6 address 2::1 64
[SwitchA-Vlan-interface2] quit
[SwitchA] interface vlan-interface 3
[SwitchA-Vlan-interface3] ipv6 address 1::1 64
```

Enable DHCPv6 relay agent and specify the DHCPv6 server address on VLAN-interface 3.

```
[SwitchA-Vlan-interface3] ipv6 dhcp relay server-address 2::2
```

2. Configure Switch A as a gateway:

Enable Switch A to send RA messages and turn on the M and O flags.

```
[SwitchA-Vlan-interface3] undo ipv6 nd ra halt
[SwitchA-Vlan-interface3] ipv6 nd autoconfig managed-address-flag
[SwitchA-Vlan-interface3] ipv6 nd autoconfig other-flag
```

Verifying the configuration

Display address information about DHCPv6 server addresses specified on the DHCPv6 relay agent on Switch A.

```
[SwitchA-Vlan-interface3] display ipv6 dhcp relay server-address all
Interface: Vlan3
Server address(es)                               Output Interface
2::2
```

Display packet statistics on the DHCPv6 relay agent.

```
[SwitchA-Vlan-interface3] display ipv6 dhcp relay statistics
Packets dropped           : 0
```


Error	:	0
Excess of rate limit	:	0
Packets received	:	14
SOLICIT	:	0
REQUEST	:	0
CONFIRM	:	0
RENEW	:	0
REBIND	:	0
RELEASE	:	0
DECLINE	:	0
INFORMATION-REQUEST	:	7
RELAY-FORWARD	:	0
RELAY-REPLY	:	7
Packets sent	:	14
ADVERTISE	:	0
RECONFIGURE	:	0
REPLY	:	7
RELAY-FORWARD	:	7
RELAY-REPLY	:	0

Configuring DHCPv6 client

Overview

Serving as a DHCPv6 client, the device only supports stateless DHCPv6 configuration, that is, the device can only obtain other network configuration parameters, except the IPv6 address and prefix from the DHCPv6 server.

With an IPv6 address obtained through stateless address autoconfiguration, the device automatically enables the stateless DHCPv6 function after it receives an RA message with the M flag set to 0 and the O flag set to 1.

Configuring the DHCPv6 client

Configuration prerequisites

To make the DHCPv6 client successfully obtain configuration parameters through stateless DHCPv6 configuration, make sure that the DHCPv6 server is available.

Configuration guidelines

- For more information about the **ipv6 address auto** command, see the *Layer 3—IP Services Command Reference*.
- H3C does not recommend enabling the DHCPv6 client and DHCPv6 server, or the DHCPv6 client and DHCPv6 relay agent on the same interface at the same time.

Configuration procedure

To configure the DHCPv6 client:

Step	Command
1. Enter system view.	system-view
2. Enable the IPv6 packet forwarding function.	ipv6
3. Enter interface view.	interface <i>interface-type interface-number</i>
4. Enable IPv6 stateless address autoconfiguration.	ipv6 address auto

Setting the DSCP value for DHCPv6 packets

An IPv6 packet header contains an 8-bit Traffic class field. This field identifies the service type of IPv6 packets. As defined in RFC 2474, the first six bits set the Differentiated Services Code Point (DSCP) value, and the last two bits are reserved. Network devices use the DSCP value as a reference to determine the packet priority for transmission.

To set the DSCP value for DHCPv6 packets:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Set the DSCP value for the DHCPv6 packets sent by the DHCPv6 client.	ipv6 dhcp client dscp <i>dscp-value</i>	Optional. By default, the DSCP value in DHCPv6 packets is 56.

Displaying and maintaining the DHCPv6 client

Task	Command	Remarks
Display DHCPv6 client information.	display ipv6 dhcp client [interface <i>interface-type interface-number</i>] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display DHCPv6 client statistics.	display ipv6 dhcp client statistics [interface <i>interface-type interface-number</i>] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display the DUID of the local device.	display ipv6 dhcp duid [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Clear DHCPv6 client statistics.	reset ipv6 dhcp client statistics [interface <i>interface-type interface-number</i>]	Available in user view

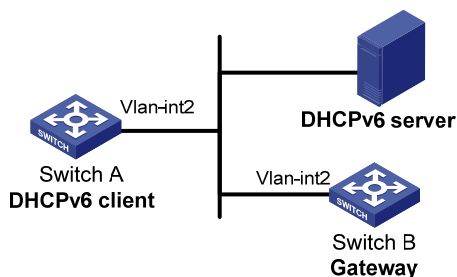
Stateless DHCPv6 configuration example

Network requirements

As shown in Figure 69, through stateless DHCPv6, Switch A obtains the DNS server address, domain name, and other information from the server.

Switch B acts as the gateway to send RA messages periodically.

Figure 69 Network diagram



Configuration procedure

- Configure Switch B:
 - # Enable the IPv6 packet forwarding function.
 - <SwitchB> system-view

```
[SwitchB] ipv6
# Configure the IPv6 address of VLAN-interface 2.
[SwitchB] interface vlan-interface 2
[SwitchB-Vlan-interface2] ipv6 address 1::1 64
# Set the O flag in the RA messages to 1.
[SwitchB-Vlan-interface2] ipv6 nd autoconfig other-flag
# Enable Switch B to send RA messages.
[SwitchB-Vlan-interface2] undo ipv6 nd ra halt
```

2. Configure Switch A:

```
# Enable the IPv6 packet forwarding function.
<SwitchA> system-view
[SwitchA] ipv6
# Enable stateless IPv6 address autoconfiguration on VLAN-interface 2.
[SwitchA] interface vlan-interface 2
[SwitchA-Vlan-interface2] ipv6 address auto
```

With this command executed, if VLAN-interface 2 has no IPv6 address configured, Switch A will automatically generate a link-local address, and send an RS message, requesting the gateway (Switch B) to reply with an RA message immediately.

Verifying the configuration

After receiving an RA message with the M flag set to 0 and the O flag set to 1, Switch A automatically enables the stateless DHCPv6 function.

Use the **display ipv6 dhcp client** command to view the current client configuration information. If the client successfully obtains configuration information from the server, the following information will be displayed.

```
[SwitchA-Vlan-interface2] display ipv6 dhcp client interface vlan-interface 2
Vlan-interface2 is in stateless DHCPv6 client mode
State is OPEN
Preferred Server:
  Reachable via address      : FE80::213:7FFF:FEF6:C818
  DUID                       : 0003000100137ff6c818
  DNS servers                : 1:2:3:5
                             1:2:4:7
  Domain names               : abc.com
                             Sysname.com
```

Use the **display ipv6 dhcp client statistics** command to view the current client statistics.

```
[SwitchA-Vlan-interface2] display ipv6 dhcp client statistics
Interface                  : Vlan-interface2
Packets Received           : 1
  Reply                    : 1
  Advertise                : 0
  Reconfigure              : 0
  Invalid                  : 0
Packets Sent               : 5
  Solicit                  : 0
  Request                  : 0
```

Confirm	:	0
Renew	:	0
Rebind	:	0
Information-request	:	5
Release	:	0
Decline	:	0

Configuring DHCPv6 snooping

A DHCPv6 snooping device does not work if it is between a DHCPv6 relay agent and a DHCPv6 server. The DHCPv6 snooping device works when it is between a DHCPv6 client and a DHCPv6 relay agent or between a DHCPv6 client and a DHCPv6 server.

You can configure only Layer 2 Ethernet ports or Layer 2 aggregate interfaces as DHCPv6 snooping trusted ports. For more information about aggregate interfaces, see *Layer 2—LAN Switching Configuration Guide*.

Overview

DHCPv6 snooping is security feature with the following functions:

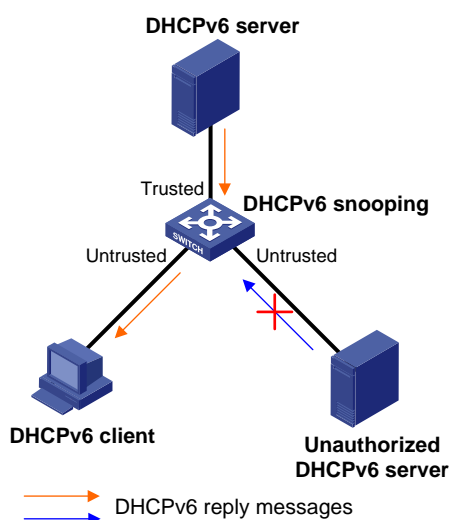
- Ensure that DHCPv6 clients obtain IPv6 addresses from authorized DHCPv6 servers.
- Record IP-to-MAC mappings of DHCPv6 clients.

Ensuring that DHCPv6 clients obtain IPv6 addresses from authorized DHCPv6 servers

If DHCPv6 clients obtain invalid IPv6 addresses and network configuration parameters from an unauthorized DHCP server, they will be unable to communicate normally with other network devices. With DHCPv6 snooping, the ports of a device can be configured as trusted or untrusted to make sure that the clients obtain IPv6 addresses only from authorized DHCPv6 servers.

- **Trusted**—A trusted port forwards DHCPv6 messages normally.
- **Untrusted**—An untrusted port discards reply messages from any DHCPv6 server.

Figure 70 Trusted and untrusted ports



A DHCPv6 snooping device's port that is connected to an authorized DHCPv6 server, DHCPv6 relay agent, or another DHCPv6 snooping device should be configured as a trusted port. The trusted port forwards reply messages from the authorized DHCPv6 server. Other ports are configured as untrusted so

that they do not forward reply messages from any DHCPv6 servers. This ensures that the DHCPv6 client can obtain an IPv6 address from the authorized DHCPv6 server only.

As shown in [Figure 70](#), configure the port that connects to the DHCPv6 server as a trusted port, and other ports as untrusted.

Recording IP-to-MAC mappings of DHCPv6 clients

DHCPv6 snooping reads DHCPv6 messages to create and update DHCPv6 snooping entries, including MAC addresses of clients, IPv6 addresses obtained by the clients, ports that connect to DHCPv6 clients, and VLANs to which the ports belong. You can use the **display ipv6 dhcp snooping user-binding** command to view the IPv6 address obtained by each client, so you can manage and monitor the clients' IPv6 addresses.

Enabling DHCPv6 snooping

To allow clients to obtain IPv6 addresses from an authorized DHCPv6 server, enable DHCPv6 snooping globally and configure trusted and untrusted ports properly. To record DHCPv6 snooping entries for a VLAN, enable DHCPv6 snooping for the VLAN.

To enable DHCPv6 snooping:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable DHCPv6 snooping globally.	ipv6 dhcp snooping enable	Disabled by default.
3. Enter VLAN view.	vlan <i>vlan-id</i>	N/A
4. Enable DHCPv6 snooping for the VLAN.	ipv6 dhcp snooping vlan enable	Optional. Disabled by default.

Configuring a DHCPv6 snooping trusted port

After enabling DHCPv6 snooping globally, you can specify trusted and untrusted ports for a VLAN as needed. A DHCPv6 snooping trusted port normally forwards received DHCPv6 packets. A DHCPv6 snooping untrusted port discards any DHCPv6 reply message received from a DHCPv6 server. Upon receiving a DHCPv6 request from a client in the VLAN, the DHCPv6 snooping device forwards the packet through trusted ports rather than any untrusted port in the VLAN, reducing network traffic.

You must specify a port connected to an authorized DHCPv6 server as trusted to make sure that DHCPv6 clients can obtain valid IPv6 addresses. The trusted port and the ports connected to the DHCPv6 clients must be in the same VLAN.

If a Layer 2 Ethernet port is added to an aggregation group, the DHCPv6 snooping configuration of the interface will not take effect until the interface quits from the aggregation group.

To configure a DHCPv6 snooping trusted port:

Step	Command	Remarks
1. Enter system view.	system-view	N/A

Step	Command	Remarks
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Configure the port as trusted.	ipv6 dhcp snooping trust	By default, all ports of the device with DHCPv6 snooping globally enabled are untrusted.

Configuring the maximum number of DHCPv6 snooping entries an interface can learn

Perform this optional task to prevent an interface from learning too many DHCPv6 snooping entries and to save system resources.

To configure the maximum number of DHCPv6 snooping entries an interface can learn:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Configure the maximum number of DHCPv6 snooping entries that the interface can learn.	ipv6 dhcp snooping max-learning-num <i>number</i>	Optional. By default, the number of DHCPv6 snooping entries learned by an interface is not limited.

Configuring DHCPv6 snooping to support Option 18 and Option 37

Option 18 is the Interface ID option and Option 37 is the Remote ID option. Upon receiving a DHCPv6 request, the DHCPv6 snooping device adds Option 18 or Option 37 into the request message before forwarding it to the DHCPv6 server.

Figure 71 Option 18 format

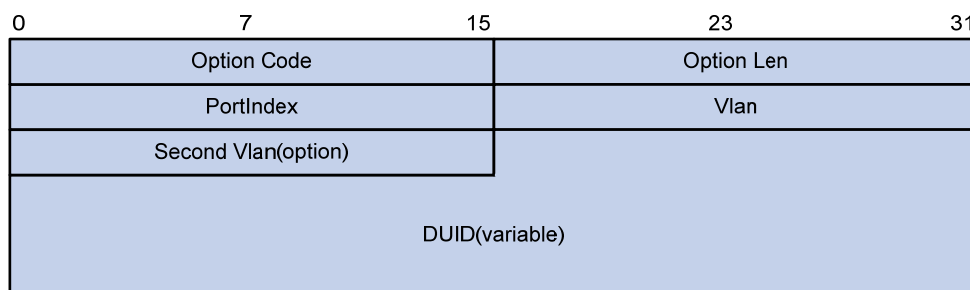
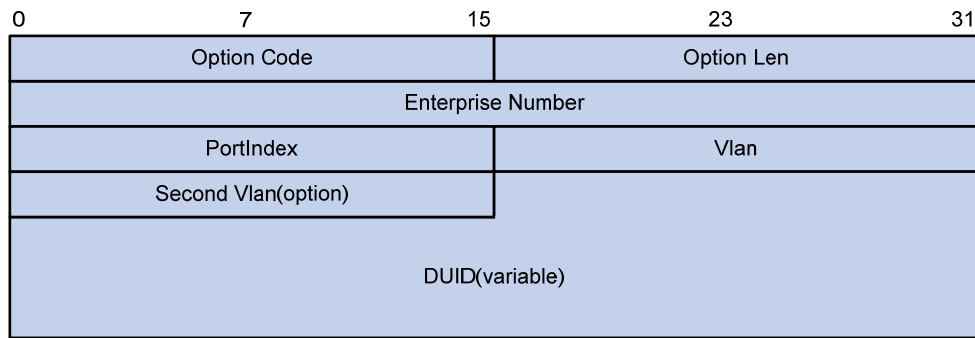


Figure 72 Option 37 format



The Second Vlan field is optional. If the received DHCPv6 request does not contain a second VLAN, Option 18 or Option 37 also does not contain it.

To configure DHCPv6 Snooping to support Option 18 and Option 37:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable DHCPv6 Snooping.	ipv6 dhcp snooping enable	Disabled by default.
3. Enter VLAN view.	vlan <i>vlan-id</i>	N/A
4. Enable DHCPv6 snooping in the VLAN.	ipv6 dhcp snooping vlan enable	Disabled by default.
5. Enter Layer 2 Ethernet port view or Layer 2 aggregate interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
6. Enable DHCPv6 snooping to support Option 18.	ipv6 dhcp snooping option interface-id enable	By default, DHCPv6 snooping does not support Option 18.
7. Configure the DUID in Option 18.	ipv6 dhcp snooping option interface-id string <i>interface-id</i>	Optional. By default, the DUID in Option 18 is the DUID of the device.
8. Enable DHCPv6 snooping to support Option 37.	ipv6 dhcp snooping option remote-id enable	By default, DHCPv6 snooping does not support Option 37.
9. Configure the DUID in Option 37.	ipv6 dhcp snooping option remote-id string <i>remote-id</i>	Optional. By default, the DUID in Option 37 is the DUID of the device.

Displaying and maintaining DHCPv6 snooping

Task	Command	Remarks
Display DHCPv6 snooping trusted ports.	display ipv6 dhcp snooping trust [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display DHCPv6 snooping entries.	display ipv6 dhcp snooping user-binding { <i>ipv6-address</i> dynamic } [{ begin exclude include } <i>regular-expression</i>]	Available in any view

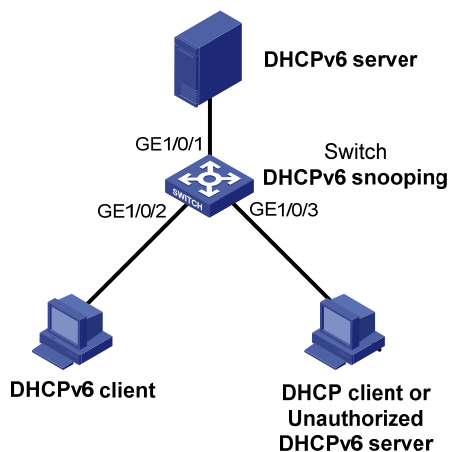
Task	Command	Remarks
Clear DHCPv6 snooping entries.	reset ipv6 dhcp snooping user-binding { ipv6-address dynamic }	Available in user view

DHCPv6 snooping configuration example

Network requirements

As shown in [Figure 73](#), Switch is connected to a DHCPv6 server through GigabitEthernet 1/0/1, and is connected to DHCPv6 clients through GigabitEthernet 1/0/2 and GigabitEthernet 1/0/3. These three interfaces belong to VLAN 2. Configure Switch to forward DHCPv6 reply messages received on GigabitEthernet 1/0/1 only and record the IP-to-MAC mappings for DHCPv6 clients.

Figure 73 Network diagram



Configuration procedure

Enable DHCPv6 snooping globally.

```
<Switch> system-view
[Switch] ipv6 dhcp snooping enable
```

Add GigabitEthernet 1/0/1, GigabitEthernet 1/0/2, and GigabitEthernet 1/0/3 to VLAN 2.

```
[Switch] vlan 2
[Switch-vlan2] port GigabitEthernet 1/0/1 GigabitEthernet 1/0/2 GigabitEthernet 1/0/3
```

Enable DHCPv6 snooping for VLAN 2.

```
[Switch-vlan2] ipv6 dhcp snooping vlan enable
[Switch] quit
```

Configure GigabitEthernet 1/0/1 as a DHCPv6 snooping trusted port.

```
[Switch] interface GigabitEthernet 1/0/1
[Switch-GigabitEthernet1/0/1] ipv6 dhcp snooping trust
```

Verifying the configuration

Connect GigabitEthernet 1/0/2 to a DHCPv6 client, GigabitEthernet 1/0/1 to a DHCPv6 server, and GigabitEthernet 1/0/3 to an unauthorized DHCPv6 server. The DHCPv6 client obtains an IPv6 address from DHCPv6 server, but cannot obtain any IPv6 address from the unauthorized DHCPv6 server. You can use the **display ipv6 dhcp snooping user-binding** command to view the DHCPv6 snooping entries on Switch.

Configuring IPv6 DNS

Overview

IPv6 Domain Name System (DNS) is responsible for translating domain names into IPv6 addresses. Like IPv4 DNS, IPv6 DNS includes static domain name resolution and dynamic domain name resolution. The functions and implementations of the two types of domain name resolution are the same as those of IPv4 DNS. For more information, see "Configuring IPv4 DNS."

Configuring the IPv6 DNS client

Configuring static domain name resolution

Configuring static domain name resolution refers to specifying the mappings between host names and IPv6 addresses. Static domain name resolution allows applications such as Telnet to contact hosts by using host names instead of IPv6 addresses.

Follow these guidelines when you configure static domain name resolution:

- A host name can be mapped to one IPv6 address only. If you map a host name to different IPv6 addresses, the last configuration takes effect.
- You can configure up to 50 mappings between domain name and IPv6 address on the switch.

To configure static domain name resolution:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Configure a mapping between a host name and an IPv6 address.	ipv6 host <i>hostname ipv6-address</i>	Not configured by default

Configuring dynamic domain name resolution

To send DNS queries to a correct server for resolution, dynamic domain name resolution needs to be enabled and a DNS server needs to be configured.

In addition, you can configure a DNS suffix that the system automatically adds to the provided domain name for resolution.

Follow these guidelines when you configure dynamic domain name resolution:

- You can configure up to six DNS servers, including those with IPv4 addresses on the switch.
- You can specify up to ten DNS suffixes on the switch.

To configure dynamic domain name resolution:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable dynamic domain name resolution.	dns resolve	Disabled by default.
3. Specify a DNS server.	dns server ipv6 <i>ipv6-address</i> [<i>interface-type interface-number</i>]	Not specified by default. If the IPv6 address of a DNS server is a link-local address, you must specify the <i>interface-type</i> and <i>interface-number</i> arguments.
4. Configure a DNS suffix.	dns domain <i>domain-name</i>	Optional. Not configured by default. Only the provided domain name is resolved.

Setting the DSCP value for IPv6 DNS packets

An IPv6 packet header contains an 8-bit Traffic class field. This field identifies the service type of IPv6 packets. As defined in RFC 2474, the first six bits set the Differentiated Services Code Point (DSCP) value, and the last two bits are reserved. Network devices use the DSCP value as a reference to determine the packet priority for transmission.

To set the DSCP value for IPv6 DNS packets:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Set the DSCP value for IPv6 DNS packets.	dns ipv6 dscp <i>dscp-value</i>	Optional. By default, the DSCP value in IPv6 DNS packets is 0.

Displaying and maintaining IPv6 DNS

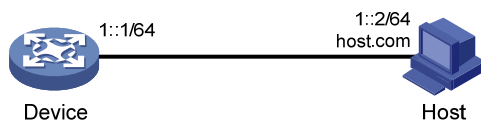
Task	Command	Remarks
Display the static IPv6 domain name resolution table.	display ipv6 host [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display IPv6 DNS server information.	display dns ipv6 server [dynamic] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display DNS suffixes.	display dns domain [dynamic] [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Display information about dynamic IPv6 domain name cache.	display dns host ipv6 [{ begin exclude include } <i>regular-expression</i>]	Available in any view
Clear information about dynamic IPv6 domain name cache.	reset dns host ipv6	Available in user view

Static domain name resolution configuration example

Network requirements

As shown in [Figure 74](#), the device wants to access the host by using an easy-to-remember domain name rather than an IPv6 address. Configure static domain name resolution on the device so that the device can use the domain name `host.com` to access the host whose IPv6 address is `1::2`.

Figure 74 Network diagram



Configuration procedure

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

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

Enable IPv6 packet forwarding.

```
[Device] ipv6
```

Use the **ping ipv6 host.com** command to verify that the device can use static domain name resolution to resolve domain name `host.com` into IPv6 address `1::2`.

```
[Device] ping ipv6 host.com
PING host.com (1::2):
 56 data bytes, press CTRL_C to break
  Reply from 1::2
  bytes=56 Sequence=1 hop limit=64 time = 3 ms
  Reply from 1::2
  bytes=56 Sequence=2 hop limit=64 time = 1 ms
  Reply from 1::2
  bytes=56 Sequence=3 hop limit=64 time = 1 ms
  Reply from 1::2
  bytes=56 Sequence=4 hop limit=64 time = 2 ms
  Reply from 1::2
  bytes=56 Sequence=5 hop limit=64 time = 2 ms
--- host.com ping statistics ---
 5 packet(s) transmitted
 5 packet(s) received
 0.00% packet loss
 round-trip min/avg/max = 1/1/3 ms
```

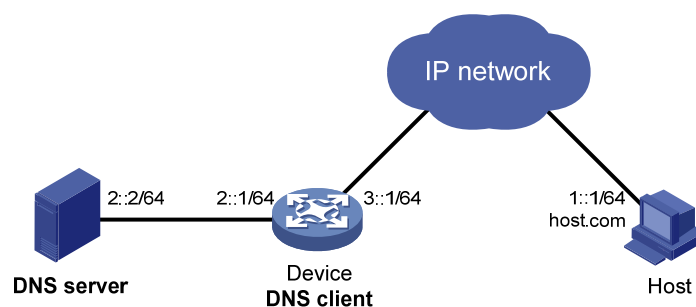
Dynamic domain name resolution configuration example

Network requirements

As shown in [Figure 75](#), the device wants to access the host by using an easy-to-remember domain name rather than an IPv6 address. The IPv6 address of the DNS server is $2::2/64$ and the server has a com domain, which stores the mapping between domain name host and IPv6 address $1::1/64$.

Configure dynamic domain name resolution and the domain name suffix com on the device that serves as a DNS client so that the device can use domain name host to access the host with the domain name host.com and the IPv6 address $1::1/64$.

Figure 75 Network diagram



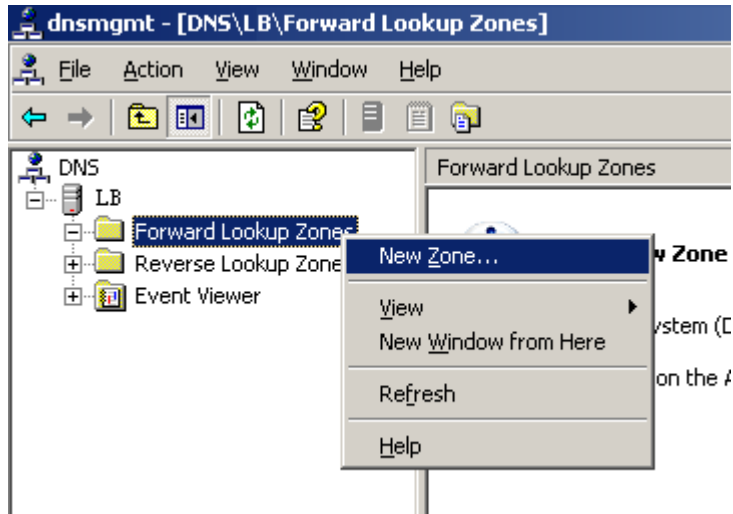
Configuration procedure

Before performing the following configuration, make sure the device and the host are accessible to each other via available routes, and the IPv6 addresses of the interfaces are configured as shown [Figure 75](#).

This configuration may vary with DNS servers. The following configuration is performed on a PC running Windows Server 2003. Make sure that the DNS server supports the IPv6 DNS function so that the server can process IPv6 DNS packets, and the interfaces of the DNS server can forward IPv6 packets.

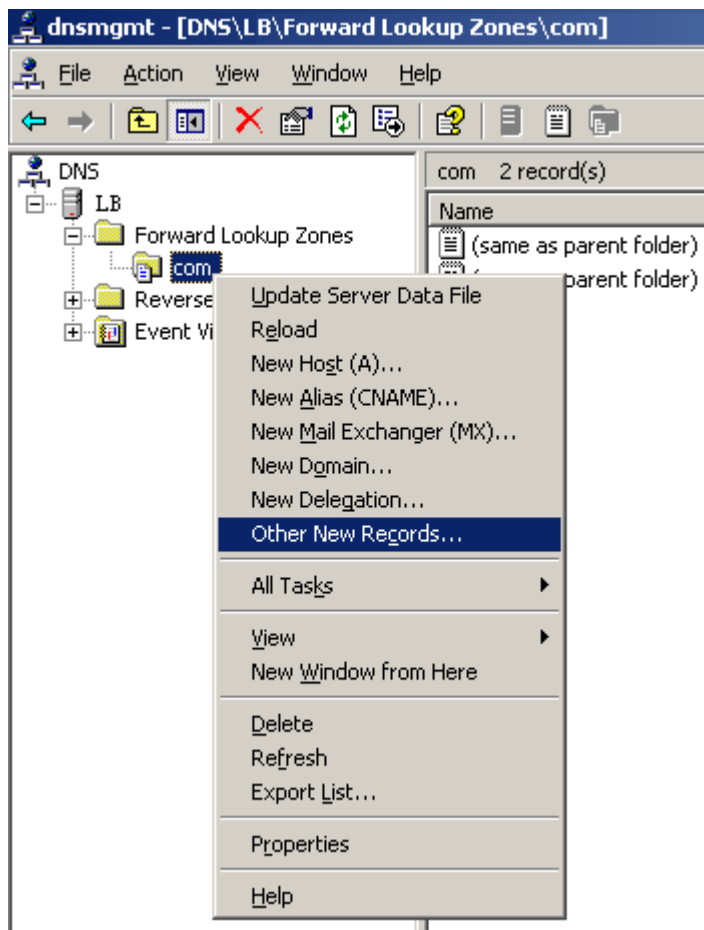
1. Configure the DNS server:
 - a. Select **Start > Programs > Administrative Tools > DNS**.
The DNS server configuration page appears, as shown in [Figure 76](#).
 - b. Right-click **Forward Lookup Zones**, select **New Zone**, and then follow the instructions to create a new zone named **com**.

Figure 76 Creating a zone



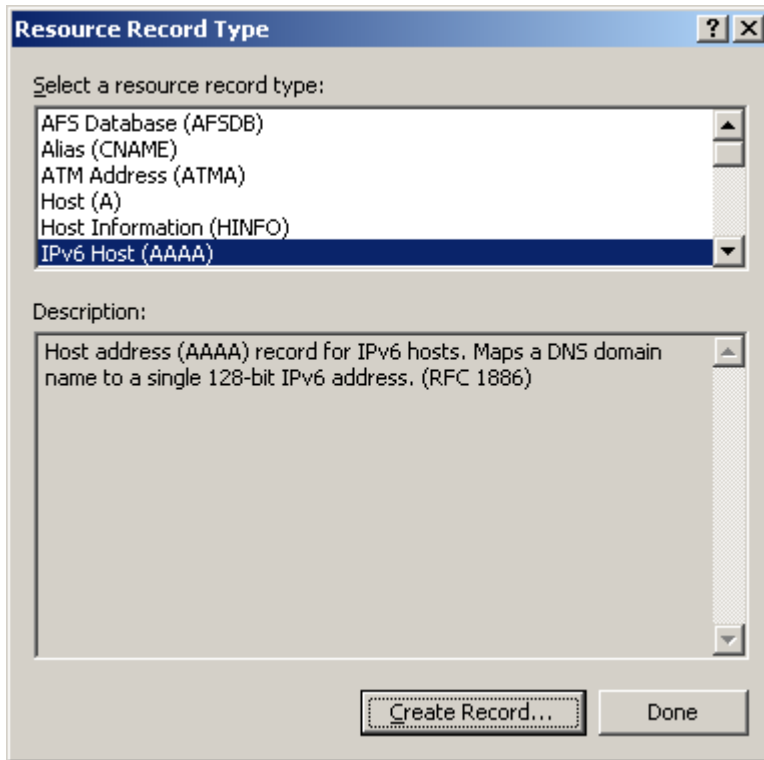
- c. On the DNS server configuration page, right-click zone **com** and select **Other New Records**.

Figure 77 Creating a record



- d. On the page that appears, select **IPv6 Host (AAAA)** as the resource record type, and click **Create Record**.

Figure 78 Selecting the resource record type



- e. On the page that appears, enter host name **host** and IPv6 address **1::1**.
 - f. Click **OK**.
- The mapping between the IP address and host name is created.

Figure 79 Adding a mapping between domain name and IPv6 address

The screenshot shows a window titled "New Resource Record" with a tab labeled "IPv6 Host (AAAA)". Inside the window, there are three text input fields. The first is labeled "Host (uses parent domain if left blank):" and contains the text "host". The second is labeled "Fully qualified domain name (FQDN):" and contains "host.com.". The third is labeled "IP version 6 host address:" and contains "1::1". At the bottom of the window, there are two buttons: "OK" and "Cancel".

2. Configure the DNS client:

Enable dynamic domain name resolution.

```
<Device> system-view
```

```
[Device] dns resolve
```

Specify the DNS server 2::2.

```
[Device] dns server ipv6 2::2
```

Configure com as the DNS suffix.

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

Verifying the configuration

Use the **ping ipv6 host** command on the device to verify that the communication between the device and the host is normal and that the corresponding destination IP address is 1::1.

```
[Device] ping ipv6 host
```

```
Trying DNS resolve, press CTRL_C to break
```

```
Trying DNS server (2::2)
```

```
PING host.com (1::1):
```

```
56 data bytes, press CTRL_C to break
```

```
Reply from 1::1
```

```
bytes=56 Sequence=1 hop limit=126 time = 2 ms
```

```
Reply from 1::1
```

```
bytes=56 Sequence=2 hop limit=126 time = 1 ms
Reply from 1::1
bytes=56 Sequence=3 hop limit=126 time = 1 ms
Reply from 1::1
bytes=56 Sequence=4 hop limit=126 time = 1 ms
Reply from 1::1
bytes=56 Sequence=5 hop limit=126 time = 1 ms
```

```
--- host.com ping statistics ---
```

```
5 packet(s) transmitted
```

```
5 packet(s) received
```

```
0.00% packet loss
```

```
round-trip min/avg/max = 1/1/2 ms
```

Index

[A](#) [B](#) [C](#) [D](#) [E](#) [I](#) [O](#) [P](#) [S](#) [T](#) [U](#)

A

- Address/prefix lease renewal, [141](#)
- Application environment of trusted ports, [69](#)
- Applying an extended address pool on an interface, [44](#)
- Applying the address pool to an interface, [147](#)
- ARP configuration examples, [7](#)
- Assigning an IP address to an interface, [21](#)

B

- BOOTP client configuration example, [81](#)

C

- Configuration guidelines, [10](#)
- Configuration procedure, [11](#)
- Configuration procedure, [96](#)
- Configuration procedure, [18](#)
- Configuration procedure, [106](#)
- Configuration restrictions, [65](#)
- Configuration restrictions, [80](#)
- Configuration restrictions and guidelines, [106](#)
- Configuring a DHCPv6 address pool, [146](#)
- Configuring a DHCPv6 snooping trusted port, [162](#)
- Configuring a static ARP entry, [3](#)
- Configuring an address pool for the DHCP server, [35](#)
- Configuring an interface to dynamically obtain an IP address through BOOTP, [81](#)
- Configuring ARP quick update, [5](#)
- Configuring basic IPv6 functions, [118](#)
- Configuring DHCP packet rate limit, [76](#)
- Configuring DHCP snooping basic functions, [72](#)
- Configuring DHCP snooping entries backup, [75](#)
- Configuring DHCP snooping to support Option 82, [73](#)
- Configuring DHCPv6 snooping to support Option 18 and Option 37, [163](#)
- Configuring DNS spoofing, [87](#)
- Configuring ICMP to send error packets, [103](#)
- Configuring ICMPv6 packet sending, [131](#)
- Configuring IPv6 ND, [122](#)

- Configuring IPv6 TCP properties, [130](#)
- Configuring multicast ARP, [5](#)
- Configuring path MTU discovery, [130](#)
- Configuring stateless DHCPv6, [142](#)
- Configuring TCP attributes, [101](#)
- Configuring the DHCP relay agent security functions, [56](#)
- Configuring the DHCP relay agent to release an IP address, [59](#)
- Configuring the DHCP relay agent to support Option 82, [59](#)
- Configuring the DHCP server security functions, [44](#)
- Configuring the DHCPv6 client, [157](#)
- Configuring the DHCPv6 relay agent, [153](#)
- Configuring the DNS proxy, [86](#)
- Configuring the IPv4 DNS client, [85](#)
- Configuring the IPv6 DNS client, [167](#)
- Configuring the maximum number of DHCPv6 snooping entries an interface can learn, [163](#)
- Configuring the maximum number of dynamic ARP entries for an interface, [4](#)
- Configuring trusted ports in a cascaded network, [70](#)
- Correlating a DHCP server group with a relay agent interface, [55](#)
- Creating a prefix pool, [146](#)

D

- DHCP address allocation, [24](#)
- DHCP client configuration example, [66](#)
- DHCP message format, [26](#)
- DHCP options, [27](#)
- DHCP relay agent configuration examples, [62](#)
- DHCP relay agent configuration task list, [54](#)
- DHCP server configuration examples, [48](#)
- DHCP server configuration task list, [34](#)
- DHCP snooping configuration examples, [77](#)
- DHCP snooping configuration task list, [72](#)
- DHCP snooping functions, [69](#)
- DHCP snooping support for Option 82, [71](#)

- DHCPv6 address/prefix assignment, [140](#)
- DHCPv6 relay agent configuration example, [154](#)
- DHCPv6 server configuration example, [149](#)
- DHCPv6 server configuration task list, [145](#)
- DHCPv6 snooping configuration example, [165](#)
- Displaying and maintaining ARP, [6](#)
- Displaying and maintaining ARP snooping, [18](#)
- Displaying and maintaining BOOTP client configuration, [81](#)
- Displaying and maintaining DHCP snooping, [77](#)
- Displaying and maintaining DHCPv6 snooping, [164](#)
- Displaying and maintaining IP addressing, [23](#)
- Displaying and maintaining IP performance optimization, [105](#)
- Displaying and maintaining IPv4 DNS, [88](#)
- Displaying and maintaining IPv6 basics configuration, [133](#)
- Displaying and maintaining IPv6 DNS, [168](#)
- Displaying and maintaining proxy ARP, [13](#)
- Displaying and maintaining the DHCP client, [66](#)
- Displaying and maintaining the DHCP relay agent, [61](#)
- Displaying and maintaining the DHCP server, [47](#)
- Displaying and maintaining the DHCPv6 client, [158](#)
- Displaying and maintaining the DHCPv6 relay agent, [154](#)
- Displaying and maintaining the DHCPv6 server, [148](#)
- Displaying and maintaining UDP helper, [107](#)
- DNS proxy configuration example, [92](#)
- Dynamic domain name resolution configuration example, [89](#)
- Dynamic domain name resolution configuration example, [170](#)

E

- Enabling client offline detection, [45](#)
- Enabling common proxy ARP, [13](#)
- Enabling DHCP, [55](#)
- Enabling DHCP, [43](#)
- Enabling DHCP starvation attack protection, [75](#)
- Enabling DHCP-REQUEST message attack protection, [76](#)
- Enabling DHCPv6 snooping, [162](#)
- Enabling dynamic ARP entry check, [4](#)
- Enabling handling of Option 82, [46](#)
- Enabling IP conflict notification, [11](#)
- Enabling local proxy ARP, [13](#)

- Enabling offline detection, [59](#)
- Enabling receiving and forwarding of directed broadcasts to a directly connected network, [100](#)
- Enabling the DHCP client on an interface, [65](#)
- Enabling the DHCP relay agent on an interface, [55](#)
- Enabling the DHCP server on an interface, [43](#)
- Enabling the DHCPv6 server, [146](#)

I

- Introduction to DHCPv6, [140](#)
- IPv6 basics configuration example, [134](#)
- IPv6 basics configuration task list, [117](#)
- IRDP configuration example, [97](#)

O

- Overview, [109](#)
- Overview, [82](#)
- Overview, [1](#)
- Overview, [12](#)
- Overview, [157](#)
- Overview, [144](#)
- Overview, [95](#)
- Overview, [10](#)
- Overview, [106](#)
- Overview, [152](#)
- Overview, [80](#)
- Overview, [18](#)
- Overview, [53](#)
- Overview, [33](#)
- Overview, [167](#)
- Overview, [19](#)
- Overview, [161](#)

P

- Protocols and standards, [143](#)
- Protocols and standards, [31](#)
- Proxy ARP configuration examples, [14](#)

S

- Setting the aging timer for dynamic ARP entries, [4](#)
- Setting the DSCP value for DHCP packets, [47](#)
- Setting the DSCP value for DHCP packets, [66](#)
- Setting the DSCP value for DHCP packets, [61](#)
- Setting the DSCP value for DHCPv6 packets, [154](#)
- Setting the DSCP value for DHCPv6 packets, [148](#)
- Setting the DSCP value for DHCPv6 packets, [157](#)
- Setting the DSCP value for DNS packets, [87](#)

Setting the DSCP value for IPv6 DNS packets, [168](#)
Specifying the source interface for DNS packets, [87](#)
Specifying the threshold for sending trap messages, [46](#)
Stateless DHCPv6 configuration example, [158](#)
Static domain name resolution configuration example, [88](#)
Static domain name resolution configuration example, [169](#)

T

Troubleshooting DHCP relay agent configuration, [63](#)
Troubleshooting DHCP server configuration, [52](#)
Troubleshooting IPv4 DNS configuration, [94](#)
Troubleshooting IPv6 basics configuration, [139](#)

U

UDP helper configuration example, [107](#)