Advanced DHCP Features

This chapter delves into advanced configurations and functionalities for both DHCPv4 and DHCPv6, enabling more granular control, enhanced security, and improved scalability for IP address management. It covers various DHCP Options, server behaviors, and relay agent capabilities beyond basic address assignment.

Feature History for Advanced DHCP Features

This table provides release and platform support information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

Table 1: Feature History for Advanced DHCP Features

Release	Feature name and description	Supported platform
Cisco IOS XE 26.2.1 ea	Advanced DHCP Features: The support for advanced DHPC features has been introduced.	Cisco C9550 Series Smart Switches
Cisco IOS XE 17.18.1	Advanced DHCP Features: Introduced the support for advanced configurations and functionalities for both DHCPv4 and DHCPv6, enabling more granular control, enhanced security, and improved scalability for IP address management. It covers various DHCP options, server behaviors, and relay agent capabilities beyond basic address assignment.	Cisco C9350 Series Smart Switches Cisco C9610 Series Smart Switches

Advanced DHCPv4 features

DHCP Options

DHCP Options are additional configuration parameters that a DHCP server can provide to a client beyond the basic IP address, subnet mask, and default gateway. These Options are defined by RFCs, for example, RFC 2132 and allow for highly customized client configurations, supporting various network services and device types.

This section explores advanced features within DHCPv4, including the use of DHCP Options to provide specific client configurations, server behaviors for handling legacy requests, and relay agent enhancements.

DHCP option 12: host name

DHCP Option 12 allows a DHCP client to send its host name to the DHCP server, and conversely, allows the DHCP server to provide a host name to the client. This is useful for network identification and management.

Configure DHCP option 12

Use this procedure to configure DHCP Option 12 in a DHCPv4 pool, allowing the server to assign a host name to a client.

This is useful for environments where client devices are expected to have dynamically assigned host names or for logging purposes.

Before you begin

A DHCPv4 pool must already be configured on the switch.

Procedure

Step 1 Enter DHCP pool configuration mode for the desired pool.

Example:

```
Device# configure terminal
Device(config)# ip dhcp pool VLAN10_POOL
```

Step 2 Configure the host name to be assigned to clients.

Example:

```
Device(dhcp-config)# option 12 ascii YOUR-HOSTNAME
Device(dhcp-config)#exit
Device(config)#end
Device#
```

Note
The host name can be a static string or derived from client information.

What to do next

Verify client host name assignment on the client device or through DHCP server logs.

DHCP Option 43: vendor-specific information

DHCP Option 43 provides a mechanism for vendors to encapsulate their own vendor-specific information within DHCP messages. This is particularly useful for provisioning devices like IP phones, wireless access points, or other network appliances that require specific configuration parameters from their manufacturer.

Configure DHCP Option 43

Use this procedure to configure DHCP Option 43 in a DHCPv4 pool. The format of the data depends on the vendor specification.

This is essential for deploying specific vendor devices that rely on DHCP for initial configuration.

Before you begin

- A DHCPv4 pool must already be configured.
- Obtain the exact format and values for Option 43 from the vendor documentation.

Procedure

Step 1 Enter DHCP pool configuration mode for the desired pool.

Example:

```
Device# configure terminal
Device(config)# ip dhcp pool VLAN10_POOL
Device(dhcp-config)#
```

Step 2 Configure the vendor-specific information.

Example:

```
Device(dhcp-config)# option 43 hex <VENDOR_CODE> <HEX_DATA>
Device(dhcp-config)#
```

The **hex** keyword allows entering hexadecimal values, while **ascii** allows ASCII strings. Replace `VENDOR_CODE` and `HEX_DATA` with actual values.

Example for Cisco IP Phone TFTP server:

```
Device(dhcp-config)# option 43 ascii "24150.10.10.10.10"
Device(dhcp-config)# exit
Device(config)# end
Device#
```

In this example, 241 is the sub-option code for TFTP server, 50 is the length of the IP address, and 10.10.10.10 is the TFTP server IP.

What to do next

Verify that the vendor device correctly interprets and applies the received Option 43 data.

DHCP Option 82: relay agent information

DHCP Option 82, also known as the Relay Agent Information Option, allows a DHCP relay agent to insert information about itself and the client into a DHCP message before forwarding it to the DHCP server. This information helps the server make more intelligent decisions about IP address assignment, such as assigning addresses based on the client's physical location or the relay agent.

Configure DHCP Option 82 - configurable Circuit ID and Remote ID

Use this procedure to configure the content of the Remote ID and Circuit ID sub-options within DHCP Option 82 on a relay agent interface.

This is crucial for granular client identification and policy enforcement on the DHCP server.

Before you begin

The interface must be configured as a DHCP relay agent with the **ip helper-address** command.

Procedure

Step 1 Enter interface configuration mode for the interface connected to the DHCP clients.

Example:

```
Device# configure terminal
Device(config)# interface Vlan222
Device(config-if)#
```

Step 2 Configure the DHCP Option 82 information for the Circuit ID and Remote ID.

Example:

```
Device(config)#int hu1/0/29
Device(config-if)#switchport
Device(config-if)#ip dhcp snooping vlan 222 information option format-type circuit-id string
 CODX_CIRCUIT_222
Device(config-if)#
```

Remote ID:

```
Device(config)#ip dhcp snooping information option format remote-id string CODX_REMOTE_9550
Device(config)#
```

You can specify the format for the Circuit ID and the Remote ID.

What to do next

Verify on the DHCP server that Option 82 data is received and correctly interpreted.

Configure DHCP Option 82 - data insertion

Use this procedure to enable the insertion of DHCP Option 82 information by the relay agent into client requests.

Before you begin

The interface must be configured as a DHCP relay agent with the **ip helper-address** command.

Procedure

Step 1 Enter interface configuration mode for the interface connected to the DHCP clients.

Example:

```
Device# configure terminal
Device(config)# interface Vlan10
```

Step 2 Enable the insertion of Option 82 information.

Example:

```
Device(config-if)# ip dhcp relay information option
Device(config-if)# end
Device#
```

By default, this command enables insertion. To disable, use **no ip dhcp relay information option** command.

What to do next

Monitor DHCP server logs for the presence of Option 82 data.

Configure DHCP Option 82 - pass through

Use this procedure to configure the DHCP relay agent to pass through DHCP Option 82 information received from other relay agents or clients, rather than stripping or modifying it.

This is important in multi-relay scenarios where Option 82 information needs to be preserved end-to-end.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
Device(config)#
```

Step 2 Enable DHCP Option 82 - pass through.

Example:

```
Device(config)# ip dhcp snooping information option allow-untrusted
Device(config)# end
Device#
```

What to do next

Verify that the DHCP server receives Option 82 data from downstream relays as expected.

DHCP Server configuration to ignore all BOOTP requests

The Bootstrap Protocol (BOOTP) is an older protocol that DHCP superseded. While DHCP is backward compatible with BOOTP, in modern networks, it may be desirable for a DHCP server to ignore BOOTP requests to reduce processing overhead or prevent unintended IP assignments to legacy devices.

Configure DHCP server to ignore BOOTP requests

Use this procedure to configure the switch, when acting as a DHCP server, to ignore incoming BOOTP requests.

This is useful in environments where BOOTP is no longer used, or where a dedicated BOOTP server handles such requests.

Before you begin

The switch must be configured as a DHCP server.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Disable the BOOTP service.

Example:

```
Device(config)# ip dhcp bootp ignore
Device(config)#
```

Step 3 Exit global configuration mode.

Example:

```
Device(config)# end
Device#
```

What to do next

Confirm that no BOOTP clients are relying on this server for IP addresses.

DHCP relay agent support on unnumbered interfaces

Traditionally, DHCP relay agents operate on interfaces configured with an IP address. However, in some network designs, interfaces might be unnumbered. That is, they borrow an IP address from another interface. Supporting DHCP relay on unnumbered interfaces allows for more flexible network topologies, especially in point-to-point links.

Configure DHCP relay agent on unnumbered interfaces

Use this procedure to configure a DHCPv4 relay agent on an unnumbered interface.

This is typically used on point-to-point links where assigning a dedicated IP address to the interface is unnecessary.

Before you begin

- The unnumbered interface must be configured to borrow an IP address from another interface. For example, **ip unnumbered Loopback0**.
- The **ip helper-address** command must be applied to the unnumbered interface.

Procedure

Step 1 Enter interface configuration mode for the unnumbered interface.

Example:

```
Device# configure terminal
Device(config)# interface GigabitEthernet1/0/2
```

Step 2 Configure the interface as unnumbered.

Example:

```
Device(config-if)# ip unnumbered Loopback0
```

Step 3 Configure the IP helper address on the unnumbered interface.

Example:

```
Device(config-if)# ip helper-address 10.0.0.10
Device(config-if)# end
Device#
```

For this step, use the IP address of your DHCP server.

What to do next

Verify DHCP client connectivity through the unnumbered interface.

DHCP syslog

DHCP Syslog refers to the logging of DHCP-related events and messages to a syslog server. This provides valuable insights into DHCP operations, including IP address assignments, lease renewals, errors, and other diagnostic information, which is crucial for network monitoring, troubleshooting, and auditing.

Information available through DHCP syslog

- IP address lease events: records when an IP address is assigned, renewed, released, or expired. This includes the client's MAC address and the assigned IP.
- DHCP pool utilization: provides notifications when DHCP pools are nearing exhaustion or are empty, indicating potential capacity issues
- DHCP Server errors: logs errors encountered by the DHCP server, such as conflicts, database issues, or communication failures
- DHCP Client activity: details client requests, offers, and acknowledgments, aiding in tracing specific client interactions
- DHCP Relay activity: logs events related to DHCP relay operations, including forwarded messages and any issues encountered

Benefits of DHCP Syslog

- Troubleshooting: quickly diagnose DHCP-related connectivity issues.
- Auditing and compliance: maintain records of IP address assignments for security and regulatory compliance
- Capacity planning: monitor pool utilization to proactively expand IP address ranges
- Security: detect unauthorized DHCP servers or suspicious client activity

System logging

DHCP syslog messages are typically enabled as part of the broader system logging configuration on Cisco devices. There isn't a dedicated DHCP Syslog command; rather, DHCP events are included in the standard syslog output based on the configured logging level. To enable logging, use commands like **logging host** *ip-address* and **logging trap** *level* in global configuration mode.

Advanced DHCPv6 features and DHCPv6 options

This section covers advanced features specific to DHCPv6, including various options for providing additional client information, bulk lease query capabilities, and enhanced relay agent functionalities.

Similar to DHCPv4, DHCPv6 utilizes options to provide clients with more than just an IPv6 address. These options are crucial for delivering comprehensive network configurations in IPv6 environments, supporting services like DNS, NTP, and other specialized information.

DHCPv6 - NIS, NISP, SNTP and Information Refresh Options

DHCPv6 can provide clients with Network Information Service (NIS), NISP (Network Information Service Plus), Simple Network Time Protocol (SNTP) server addresses, and information refresh parameters. These options ensure clients have access to essential network services.

Configure DHCPv6 NIS, NISP, SNTP, and Information Refresh Options

Use this procedure to configure DHCPv6 options for NIS, NISP, SNTP, and information refresh within a DHCPv6 pool.

Before you begin

A DHCPv6 pool must already be configured.

Procedure

Step 1 Enter DHCPv6 pool configuration mode for the desired pool.

Example:

```
Device# configure terminal
Device(config)# ipv6 dhcp pool testv6
Device(config-dhcpv6)#
```

Step 2 Configure NIS server addresses.

Example:

```
Device(config-dhcpv6)# nis server 2001:DB8::10
```

Step 3 Configure NISP server addresses.

Example:

```
Device(config-dhcpv6)# nisp server 2001:DB8::11
```

Step 4 Configure SNTP server addresses.

Example:

```
Device(config-dhcpv6)# sntp server 2001:DB8::12
```

Step 5 (Optional) Configure the information refresh time.

Example:

```
Device(config-dhcpv6)# information refresh 86400
```

This specifies how often clients should refresh their non-address configuration information.

Step 6 Exit DHCPv6 pool configuration mode and global configuration mode.

Example:

```
Device(config-dhcpv6)# exit
Device(config)# end
```

Clients will receive the configured NIS, NISP, and SNTP server addresses, and the specified information refresh time.

What to do next

Verify client configuration and service accessibility.

DHCPv6 bulk lease query - RFC 5460

DHCPv6 Bulk Lease Query (BLQ), defined in RFC 5460, provides a mechanism for a DHCPv6 server to query a client for a bulk list of its active leases. This is particularly useful for network management systems, lawful intercept, or other applications that require a comprehensive view of IP address assignments without querying individual clients.

Benefits of BLQ

- Centralized lease management: Network administrators can obtain a complete picture of active leases from a single point.
- Enhanced monitoring: facilitates real-time tracking of IP address usage
- Troubleshooting: aids in identifying rogue devices or unexpected lease activity

How BLQ works

This differs from standard DHCPv6 interactions where clients only request or renew their own leases.

Summary

The BLQ mechanism is typically implemented on DHCPv6 servers and clients, rather than directly configured on relay agents or intermediate devices.

The process distinguishes bulk lease query behavior from standard DHCPv6 interactions where clients request or renew only their own leases.

Workflow

A BLQ-capable DHCPv6 client, upon receiving a BLQ message from a server, responds with a list of its current DHCPv6 leases.

While your switches can act as DHCPv6 servers, the direct configuration for initiating or responding to bulk lease queries is typically part of the DHCPv6 server's internal logic and not a user-configurable CLI command on the switch itself. Network management applications would interact with the server using this protocol.

DHCPv6 Relay Agent notification for prefix delegation

When a DHCPv6 client requests a prefix delegation (PD) from a DHCPv6 server, the relay agent can be configured to notify other network elements, such as routing protocols, about the delegated prefix. This allows

for dynamic routing updates based on delegated prefixes, simplifying network management for service providers.

Configure DHCPv6 Relay Agent notification for prefix delegation

Use this procedure to configure the DHCPv6 relay agent to send notifications for prefix delegation.

Before you begin

The interface must be configured as a DHCPv6 relay agent.

Procedure

Step 1 Enter interface configuration mode for the interface connected to the DHCPv6 clients.

Example:

```
Device# configure terminal
```

Step 2 Enable DHCPv6 relay agent notification for prefix delegation.

Example:

```
Device(config)#ipv6 dhcp pool testv6
Device(config-dhcpv6)# ipv6 dhcp relay prefix-delegation pool
Device(config-dhcpv6)# exit
Device(config)# end
Device#
```

What to do next

Verify that routing protocols or other services are receiving and acting upon these notifications.

DHCPv6 Relay source

When a DHCPv6 Relay Agent forwards a client's request, it typically uses the interface's IPv6 address as the source. However, it can be beneficial to configure a specific source address, such as a loopback interface address, for consistency or to ensure reachability even if the physical interface changes.

Configure DHCPv6 Relay source

Use this procedure to configure a specific source address for DHCPv6 Relay messages.

This is useful for ensuring that the DHCPv6 server always sees a consistent source address from the Relay, regardless of which physical interface the message egresses.

Before you begin

- The interface must be configured as a DHCPv6 Relay Agent.
- A loopback interface with an IPv6 address is recommended as the source.

Procedure

Step 1 Enter interface configuration mode for the interface connected to the DHCPv6 clients.

Example:

```
Device# configure terminal
Device(config)# interface Vlan20
```

Step 2 Configure the DHCPv6 relay source interface.

Example:

```
Device(config-if)# ipv6 dhcp relay source-interface Loopback0
Device(config-if)# end
Device#
```

Replace *Loopback0* with the actual interface name to be used as the source.

What to do next

Verify on the DHCPv6 server that the source address of received relay messages is the configured loopback address.

DHCPv6 Server Stateless Address Autoconfiguration

In IPv6, Stateless Address Autoconfiguration (SLAAC) allows devices to generate their own IPv6 addresses. DHCPv6 SLAAC complements SLAAC by providing additional network configuration information like, DNS server addresses or domain names without assigning the IPv6 address itself. This is achieved throug `managed-config-flag` and `other-config-flag` in Router Advertisements (RAs).

Configure DHCPv6 Server Stateless AutoConfiguration

Use this procedure to configure a DHCPv6 server to provide stateless configuration information.

This is a common deployment model for IPv6, where SLAAC handles address assignment, and DHCPv6 provides supplementary details.

Before you begin

- Configure the interface with an IPv6 address
- Enable Router Advertisements

Procedure

Step 1 Enter interface configuration mode for the interface where clients will receive stateless configuration.

Example:

```
Device# configure terminal
Device(config)# interface Vlan20
```

Step 2 Enable Router Advertisements and set the other-config-flag.

Example:

```
Device(config-if)# ipv6 nd other-config-flag
```

Alternatively, to force stateful DHCPv6 address assignment, set the `managed-config-flag`:

```
Device(config-if)# ipv6 nd managed-config-flag
```

The `other-config-flag` tells clients to use DHCPv6 for non-address configuration.

Step 3 Associate a DHCPv6 pool for stateless configuration, if not already done.

Example:

```
Device(config-if)# ipv6 dhcp server VLAN20_IPV6_POOL
Device(config-if)# end
Device#
```

This pool contains the non-address options like DNS servers.

What to do next

Verify client IPv6 configuration, including DNS servers.

VRF-aware DHCPv6 Server and Relay for IANA and prefix delegation

Virtual Routing and Forwarding (VRF) allows a single router to host multiple independent routing tables. VRF-aware DHCPv6 enables DHCPv6 server and relay functionalities to operate within specific VRF instances, providing isolated and secure IP address management for different tenants or network segments. This applies to both IPv6 address assignment (IANA - Identity Association for Non-temporary Addresses) and prefix delegation (PD).

Configure VRF-aware DHCPv6 Server and Relay for IANA and prefix delegation

Use this procedure to configure VRF-aware DHCPv6 Server or Relay functionalities.

This is critical in multi-tenant environments or for segmenting network services where each segment requires its own DHCPv6 service.

Before you begin

VRF instances must be configured, and interfaces must be assigned to the respective VRFs.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal Device(config)#
```

Step 2 Define the DHCPv6 pool within the VRF context for Server functionality.

Example:

```
Device(config)# vrf definition RED
Device(config-vrf)# add
Device(config-vrf)# address-family ipv6
Device(config-vrf-af)# exit
Device(config-vrf)# exit
Device(config)# ipv6 dhcp pool testv6
Device(config-dhcpv6)# vrf RED
Device(config-dhcpv6)# address prefix 2001:DB8:1:1::/64
Device(config-dhcpv6)# dns-server 2001::1
Device(config-dhcpv6)#
```

Step 3 Enter interface configuration mode for the interface within the VRF.

Example:

```
Device(config)# interface Vlan30
Device(config-if)# vrf forwarding VRF_TENANT_A Device
Device(config-if)# ipv6 address 2001:DB8:1:1::1/64
```

This interface is associated with *VRF_TENANT_A*.

Step 4 Enable DHCPv6 server on the interface for IANA address assignment.

Example:

```
Device(config-if)# ipv6 dhcp server VLAN30_IPV6_POOL
```

Step 5 If acting as a Relay, then configure DHCPv6 Relay within the VRF.

Example:

```
Device(config)# interface HundredGigE1/2/0/21
Device(config-if)# switchport access vlan 2268
Device(config-if)# exit
Device(config)# interface Vlan2268
Device(config-if)# vrf forwarding CODX_DHCPV6_PD_VRF
Device(config-if)# no ip address
Device(config-if)# ipv6 address 2001:DB8:2268::1/64
Device(config-if)# ipv6 enable
Device(config-if)# ipv6 nd managed-config-flag
Device(config-if)# ipv6 nd other-config-flag
Device(config-if)# ipv6 dhcp relay destination global 138::2 sourceaddress 138::1
```

Replace *2001:DB8:2268::1* with the DHCPv6 Server address.

What to do next

Verify DHCPv6 lease assignments or relay operations within the VRF context.