

DHCP and IPoE Subscriber Management

- Feature Summary and Revision History, on page 1
- Feature Description, on page 2
- Configuring the DHCP and IPoE Subscriber Management Feature, on page 11
- DHCPv6 Raw Option Support , on page 17
- IPv6 Class Configuration and Static IP Allocation Support, on page 20
- Leased IP Hold Time, on page 26
- DHCP IP Lease Reservation, on page 30
- Enhanced support for RADIUS attributes, on page 32

Feature Summary and Revision History

Summary Data

Table 1: Summary Data

Applicable Product(s) or Functional Area	cnBNG
Applicable Platform(s)	SMI
Feature Default Setting	Disabled - Configuration Required
Related Changes in this Release	Not Applicable
Related Documentation	Not Applicable

Revision History

Table 2: Revision History

Revision Details	Release
Enhancement introduced: • Stateful-IPv6-Address-Pool attribute • Delegated-IPv6-Prefix-Pool	2025.02.0
Introduced support for IPv6 Class configuration and Static IP allocation.	2025.01.0
Introduced support for DHCPv6 Raw Option.	2024.04.0
Introduced support for the Leased IP Hold Time feature.	2022.02.0
Enhancement Introduced: The IPoE (DHCP) feature is NSO-integrated.	2021.04.0
First introduced.	2021.01.0

Feature Description

Note This feature is Network Services Orchestrator (NSO) integrated.

A session represents the logical connection between the customer premise equipment (CPE) and the network resource. To enable a subscriber to access the network resources, the network has to establish a session with the subscriber. The Cloud Native Broadband Network Gateway (cnBNG) supports the following subscriber session types:

- IPoE (DHCP)
- PPP (PPPoE)

  For more information, see PPPoE Subscriber Management.

In an IPoE subscriber session, subscribers run IPv4 or IPv6 on the CPE device and connect to the BNG through a Layer-2 aggregation or Layer-3 routed network. The IP subscriber sessions that connect through a Layer-2 aggregation network are called L2-connected and sessions that connect through routed access network are called L3-connected or routed subscriber sessions. IPoE subscriber sessions are always terminated on BNG and then routed into the service provider network. IPoE relies on DHCP to assign the IP address.

On the BNG, the DHCPv4 or DHCPv6 trigger creation of these subscribers based on the First-Sign-Of-Life (FSOL) protocol. The IP sessions to the CPE can be either:

- Single stacked, that is, running only IPv4 or IPv6

- Dual stacked, that is, running both IPv4 and IPv6

The DHCP runs as a pod to handle the FSOL for the IPoE subscribers. It handles the DHCP packet encode and decode, IP address assignment, DHCP FSM handling, and DHCP feature and rule application for the IPoE sessions. The DHCP module handles both DHCPv4 and DHCPv6 control packets to bring up corresponding address family interface (AFI).

Note In this release, only the DHCP server mode functionality is supported.

A common DHCP module handles the DHCP finite state machines (FSM) for both 5G subscribers (in SMF service) and wireline subscribers in the cnBNG. The network function (NF) specific DHCP module handles the NF specific functionality.

DHCP and IPoE Functionalities

The DHCP and IPoE Subscriber Management feature supports the following functionalities:

DHCP Server

The cnBNG CP implementation supports the DHCPv4 server mode. The DHCP server FSM handles the DHCP packets from client, IP allocation, and IP lease management.

The FSM handles the following Rx control packets:

- Discover
- Request (DORA request and renew request)
- Decline
- Inform
- Release

The DHCP server FSM sends the following control packets to the client based on the FSM states and events:

- Offer
- Ack (DORA Ack, Renew Ack and Inform Ack)
- Noack

The DHCP server implementation associates a DHCP profile to a group of subscribers. This server implementation supports the following functionalities:

- IP address allocation for the client from the configured pool in the DHCP profile.
- IP address lease allocation based on DHCP profile configuration.
- Passing Host configurations to the client using the following configurable DHCP options in the DHCP profile:
  - IP subnet mask (Option 1)
  - Boot filename (Option 67)

- Domain name (Option 15)
- NetBIOS node type (Option 46)
- NetBIOS name server (Option 44)
- Domain name server (Option 6)
- Default router (Option 3)
- Time server (Option 4)

Processing Option 82

cnBNG supports Option 82, which is the relay agent information option to figure out the sub-options. The various sub-options that the DHCP processes are:

- Circuit ID (Sub option 1)
- Remote ID (Sub option 2)

The circuit ID and remote ID field is passed to the Session Manager during session start trigger and the same is used for north-bound interactions.

DHCPv4 RADIUS Proxy

The cnBNG CP supports DHCP IPv4 RADIUS proxy for RADIUS-based authorization of DHCP leases. This is a RADIUS-based address assignment mechanism in which a DHCP server authorizes remote clients and allocates IP addresses, based on replies from a RADIUS server.

These are the steps involved in the address assignment mechanism:

- The DHCP server sends the DHCP client information to the RADIUS server.
- The RADIUS server returns all required information, primarily IPV4 address, to the DHCP server in the form of RADIUS attributes. The subnet mask is derived from the CP based on the static pool configuration. The IPv4 address sent from the RADIUS must be part of the static pool associated to the UP.
- The DHCP server translates the RADIUS attributes into DHCP options and sends this information back in a DHCP Offer message to the DHCP client.

If the IETF attribute, such as Framed-IP-Address is received from the RADIUS server, and if it is present in the user profile, then this attribute is used instead of allocating the IP address from the configured pool. The basic attributes that can come from the RADIUS server that are relevant for DHCP server options are:

- Framed IPv4 Address
- IPv4 Subnet Mask (derived in the CP from the static pool configuration)
- IPv4 Default gateway (derived in the CP from the static pool configuration)

Apart from these attributes, the dhcp-class name and address pool name attribute also can come from RADIUS. If the RADIUS sets the address pool name, then it uses this for IP allocation instead of the pool that is specified as part of the DHCP profile.

If the RADIUS server sends the dhcp-class attribute to the DHCP server, then that attribute value is used to decide other configuration parameters in the reply that is to be sent to the DHCP client. For example, if the

DHCPv4 server profile has both Class A and Class B in it, and if RADIUS server sends a reply to the DHCP server with the class name as 'B', then Class B is used to send the options back to the DHCP client. Classes can be defined under DHCP profile. The parameters and options that can be configured under DHCP profile can be configured under class also.

Additional RADIUS server attributes are allowed, but not mandatory. If a RADIUS server user profile contains a required attribute that is empty and is not available via configuration as well, the DHCP server does not generate the DHCP options.

DHCPv6 Local Server for IPv6 Subscribers

The DHCPv6 server assigns IPv6 address and prefix and other configuration attributes (such as domain name, the domain name server address and SIP servers and so on) to requesting clients. On receiving a valid request, the server assigns the client IPv6 address or prefix, a lease for the assigned IPv6 address or prefix and other requested configuration parameters. The DHCP server FSM is implemented to handle the address allocation and lease management. The FSM would handle the following control packets from the client:

- Solicit
- Request
- Renew
- Rebind
- Decline
- Information-Request
- Release

The DHCPv6 server FSM sends the following control packets to the client based on the FSM states and events:

- Advertisement
- Reply (SARR Reply, Release Reply, Renew Reply, Rebind Reply and Information request Reply)
- Relay-Reply

The DHCPv6 server implementation associates a DHCPv6 profile to a group of subscribers. The server implementation caters to the following functionalities:

- IANA address and IAPD address allocation for the client from configured pool in DHCPv6 profile.
- IANA and IAPD address lease allocation based on DHCPv6 profile configuration.
- Passing Host configurations to client using below configurable DHCP options in DHCP profile
    - AFTR support (Option 64)
    - Preference option (Option 7)
    - Domain list (Option 24)
    - DNS server IPv6 address (Option 23)

The DHCPv6 server sends the following options to the Policy plane:

- interface-id (DHCP Option 18)

- remote-id (DHCP Option 37)
- vendor-class (DHCP Option 16)
- user-class (DHCP Option 15)
- client-id(DHCP Options 1)

DHCPv6 Server - Prefix Delegation

The DHCPv6 Prefix Delegation feature enables the DHCPv6 server to hand out network address prefixes to the requesting clients. The clients use these network prefixes to assign /128 addresses to the hosts on their network. The RFC-3633 and RFC-3769 is supported for prefix delegation. The DHCPv6 Prefix Delegation feature is enabled by default for cnBNG DHCPv6 server. No other configuration is required to enable the prefix delegation. The DHCPv6 option OPTION_IA_PD (25) and OPTION_IAPREFIX (26) support to meet the prefix delegation requirement.

Note

- Only one delegated prefix per subscriber and client is supported.
- Only one OPTION_IAPREFIX is supported under one OPTION_IA_PD (25).

The cnBNG allocates addresses from the prefix pool configured under the DHCP profile.

DHCPv6 Server - Address Assignment

The DHCPv6 Address Assignment feature enables the DHCPv6 server to hand out /128 addresses to the clients. The cnBNG DHCPv6 server implementation supports the DHCPv6 OPTION_IA_NA(3) and OPTION IAADDR(5) to enable address assignment to the client.

Note

- Only one delegated prefix per subscriber and client is supported.
- Only one OPTION_IAPREFIX is supported under one OPTION_IA_PD (25).

The cnBNG allocates addresses from the prefix pool configured under the DHCP profile.

Prefix and Address Pool Support for IPv6

The cnBNG supports the configuring of the DHCPv6 address and prefix pool and associating it to the DHCPv4 and DHCPv6 server profiles. The address and prefix ranges is under the pool. cnBNG also supports downloading of the address and prefix pool name via the user profile on a per subscriber basis. The pool name downloaded via user profile is given priority over the pool name association via the DHCPv6 profile.

DHCPv6 Server with RADIUS-based Address Assignment

The cnBNG supports RADIUS-based address assignment, that is, the IANA address is downloaded as part of the user profile and is allocated to the client. Address from the user profile is given priority over the local configuration.

DHCPv6 Server with RADIUS-based Prefix Delegation

The cnBNG supports RADIUS-based prefix assignment, that is, the IAPD address is downloaded as part of the user profile and is allocated to the client. The delegated prefix from the user profile is given priority over the local configuration.

DHCPv6-provided IPv6 address of DNS server for IPv6 Subscribers

The cnBNG CP DHCPv6 server implementation supports the provision of DNS server information to clients via the DNS option (23). It supports a configuration of up to 8 DNS server ipv6 addressees via the DHCPv6 profile. The DHCPv6 server information is downloaded via the user profile on a per subscriber basis. The per subscriber DNS information in the user profile is given priority over the profile configuration.

DHCPv4 DHCPv6 Lease Timeout

The cnBNG CP provides the configuration to set the lease value under the DHCPv4 and DHCPv6 profile. This configuration determines the lease for the IP addresses allocated to the clients.

For DHCPv4 clients, the lease is set in the address time (T) option (option 51). By default, the renewal time is set as (½) * T [option 58] and rebinding time is set as (7/8) * T [option 59]. For DHCPv6 client, the lease is populated in the IA address and IA prefix option for the respective address types. By default, preferred time is set as 0.5 * T and valid time T2 is set as 0.8 * T. By default, renewal time (T1) is set as 0.5 * T and rebinding time T2 is set as 0.8 * T in OPTION_IA_PD.

The cnBNG CP tracks the lease time allocated to the clients. Ideally the client should renew (Renew request) the lease at T1 to extend the lease. If renew is failing, the client uses the rebind (broadcast request message for DHCPv4 and rebind message for DHCPv6). If the cnBNG CP does not receive the lease renewal request from the client, the lease times out after T and the corresponding address is released to the pool and removed from the client session. This can lead to an update or disconnect to the Session Manager based on the other address states. The lease timeout is applicable to both IPv4 and IPv6 addresses.

IPv6 IPoE Sessions

The IPv6 subscribers run the IPv6 from the CPE device to the BNG router and are created using the DHCPv6 protocol. The IPv6 subscribers natively run IPv6 on the CPE device and are connected to the router via a Layer-2 network or through Layer-2 aggregation.

The IPv6 subscribers are supported when they are directly connect to the cnBNG UP or via a Layer-2 aggregator. The cnBNG CP DHCPv6 server treats only DHCPv6 SOLICIT message from the subscriber / client as FSOL (First Sign Of Life) packet in case of IPoE and initiates the subscriber session creation.



Note Routed subscribers are not supported.

Dual Stack IPv6/IPv4 over IPoE

The cnBNG CP supports dual-stack IPoE subscribers, that is, both IPv4 and IPv6 address allocation for the same subscriber. In this release, cnBNG supports up to one IPv4 address, one IANA address, and one IAPD address.

Subscriber Termination over Non-default VRF

The cnBNG CP DHCPv4 and DHCPv6 servers are VRF aware. The DHCPv4 and DHCPv6 servers support the access interface in either default VRF or non-default VRF. The following table shows the VRF combination supported by DHCPv6 server.

Table 3: DHCP Supported VRF Combinations

Client Access Interface	Subscriber Interface	DHCPv6 Supported
Default VRF	Default VRF	Supported
Default VRF	Non-default VRF	Supported
Non-default VRF	Non-default VRF	Supported

DHCPv4 Raw Option Support

The cnBNG DHCP Profile configuration enables the operator to configure specific DHCPv4 options, under the DHCPv4 profile. The option value can range from 1 to 255. The option value can be either an ascii string or a hexadecimal string.

DHCPv4 and DHCPv6 Class Support

The cnBNG DHCP Profile configuration enables the operator to configure classes of DHCP options and to selectively associate them during the session setup. The DHCP Options class are selected based on certain matching DHCP options received from access network against the configured class key parameters. The DHCP Options class can also be selected based on the class name received from Policy plane. The priority is always given to the DHCP class name that the Policy plane provides. However, if the Policy plane does not provide a class name, then class selection depends on the operator-configured key parameters. The operator can configure multiple DHCP option classes for DHCPv4 and DHCPv6 separately.

The DHCP Profile consist of profile elements. Each of the DHCPv4 and DHCPv6 profiles contain the ‘default’ DHCP options list and zero or more classes of DHCP options of corresponding DHCP version.

The DHCPv4 and DHCPv6 Options Class contains a list of DHCP options and the “Match-Info” holds the information about the keys to be matched to select that class. The operator can also specify under Match-Info” the class selection that should match ‘any’ or ‘all’ the key parameters of that class.

If the DHCP Option class does not match an ongoing session or any requested DHCP Options is not found in the selected class, then the requested option is selected from the ‘default’ DHCP Options of that profile.

How it Works

This section provides a brief of how the DHCP and IPoE Subscriber Management feature works.

Call Flows

This section includes the following call flow.

cnBNG IPoE Call Flow

For IPoE session establishment, the BNG User Plane (UP) sends the DHCP packets to the BNG Control Plane (CP) using the GTP-U protocol. The following figure shows the DHCP packet call-flow and session programming between the BNG-UP and BNG-CP for IPoE session establishment.

Figure 1: cnBNG IPoE Call Flow

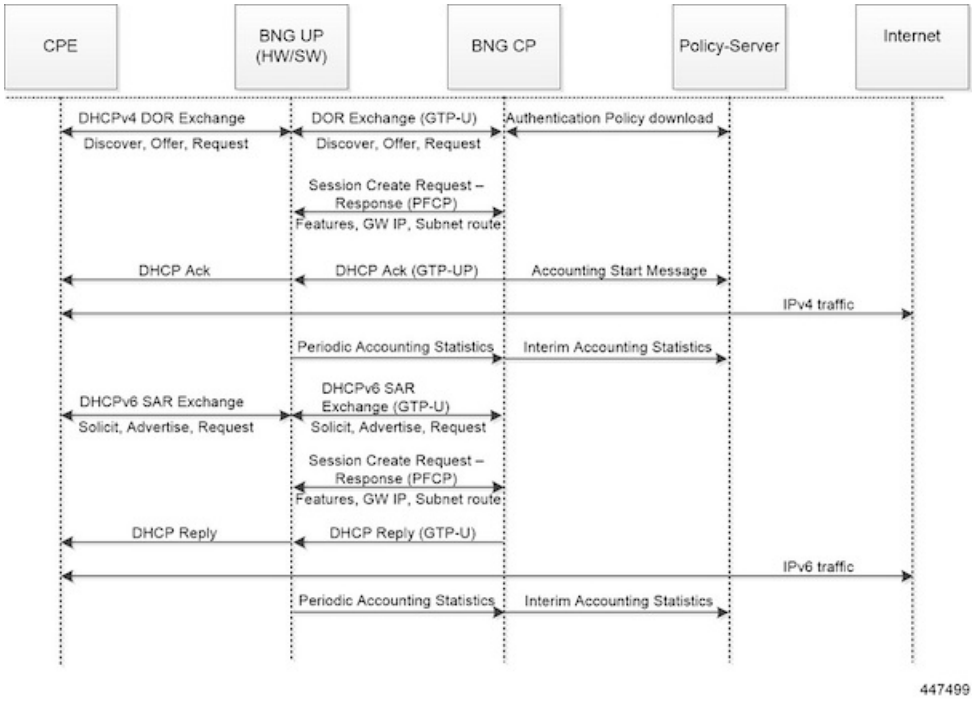


Table 4: cnBNG IPoE Call Flow Description

Step	Description
1	The subscriber running IPv4 or IPv6 stack on the CPE device connects to the BNG-UP via DHCPv4, DHCv6, or DHCPv4 and DHCPv6.
2	The BNG-UP forwards the DHCP(v4/v6) request packets received from the CPE to the BNG-CP over the GTPU protocol. It then returns the DHCP response packets received from the BNG-CP to the CPE device.
3	The BNG-CP performs the subscriber authentication via the Policy Server before establishing a subscriber session on the BNG-UP.
4	After the BNG-CP successfully establishes a session on the BNG-UP, the BNG-UP initiates the Accounting Start and trigger Session Establishment Success (DHCPv4 Ack / DHCPv6 Reply) message towards the CPE via the BNG-UP.
5	The subscriber on the CPE device initiates the data traffic (DHCPv4 / DHCPv6) via the BNG-UP or BNG-CP towards the Internet.
6	The BNG-UP forwards the periodic accounting information to the BNG-CP and the BNG-CP triggers periodic accounting towards the Policy server.

Standard Compliance

The DHCP and IPoE Subscriber Management feature caters to the DHCP server requirements only. The DHCP Server implementation is aligned with the following standards:

- RFC 2131 Dynamic Host Configuration Protocol
- RFC 2132 DHCP Options and BOOTP Vendor Extensions [Subset of options]
- RFC 3046 DHCP Relay Agent Information Option
- RFC 3004 The User Class Option for DHCP
- RFC 3315 Dynamic Host Configuration Protocol for IPv6 (DHCPv6)
- RFC 3633 IPv6 Prefix Options for Dynamic Host Configuration Protocol(DHCP)version 6
- RFC 3646 DNS Configuration options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)
- RFC 4649 Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Relay Agent Remote-ID Option
- RFC 6334 Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Option for Dual-Stack Lite

Limitations and Restrictions

The DHCP and IPoE Subscriber Management feature has the following limitations in this release:

- DHCPv6 addresses and prefixes do not get released at IPv6CP disconnect.
- For DHCPv4 sessions, subnet mask and default gateway are derived from the IPAM pool configuration and IP pool split logic. The first subnet route, subnet mask, and default gateway IP is derived from the IPAM and pushed to the UP for each chunk of the pool. Subnet mask and default gateway cannot be assigned via the AAA configuration.
- For DHCPv4 sessions, subnet selection is not supported. The IP is selected from the mapped IP pool. Subnet selection cannot be controlled via the AAA gateway IP, giaddr, or subnet selection suboption.
- For DHCPv4 sessions, requested IP option (option 50) that helps in requesting specific IP is not supported. However on client reboot (discover in bound state), the already assigned IP is retained.
- DHCP Inform packet and DHCPv6 Information Request packet handling for unbound sessions are not supported. That is, the client cannot get only the host configurations without requesting for IP assignment via BNG.
- For DHCPv4 session, broadcast flag check, and discovery, offer, request, and acknowledgement (DORA) unicast is not supported.
- If DHCP client initiated packet options like requested options (option 55 for IPv4, ORO option 6 for IPv6), circuit-id, remote-id, user class, vendor class changes in the packet over the session lifecycle, the cnBNG server behaviour is not defined. cnBNG assumes that the client will not change these options over the lifecycle of session. The client should also maintain the same values for attributes like remote-id, vendor class, user class for both IPv4 and IPv6 afi (AFI). In case these value are required to be changed, it is recommended to clear the session and bring it up again.
- Client reboot scenarios do not tear down the session in cnBNG in the following scenarios: If the Discover message is received in the Bound state or Solicit message is received for the already bound IANA, cnBNG does not tear down the existing session. Instead, the already allocated IP is assigned to the subscriber.

In this case, fresh lease is assigned to the client. This is a difference in behaviour from physical BNG where on receiving Discover message in Bound state, IPv4 stack is brought down and new IP is assigned.

- No parity support for RADIUS attribute formatting with ASR 9000. The supported RADIUS attribute list and formatting would be updated based on feedback from customer. For example, some attributes like remote-id format is different for IPv4 and IPv6 clients. Hence, the value going to the Policy Plane differs based on whether the IPv4 or IPv6 afi comes up first.
- Change of Authorization (CoA) for DHCP consumed RADIUS attributes are not supported.
- RFC recommended DHCPv4/v6 packet validations are not supported.
- A common DHCP class attribute is used for class specification for DHCPv4 and DHCPv6 stack via AAA attribute. The attribute is dhcp-class.
- Framed route is not supported.
- Manual pod restart is not supported or entertained. Pod restart can lead to inconsistencies between the CP pods with regard to session count and session state. To recover the inconsistent sessions, the **clear** command must be used explicitly.
- After a subscriber is UP, if the subscriber is deleted from the cnBNG CP—for example, due to an administrative clear or a Pod event—the subscriber is not notified.

  cnBNG CP deletion does not proactively notify the client. During a subsequent renew, cnBNG returns the DHCPNAK message for IPv4 or NoBinding message for IPv6, enabling the client to renegotiate immediately rather than waiting for the lease to expire.

Configuring the DHCP and IPoE Subscriber Management Feature

This section describes how to configure the DHCP and IPoE Subscriber Management feature.

Configuring the DHCP and IPoE Subscriber Management feature involves the following steps:

1. Configure the IPv4 DHCP Profile
2. Configure the IPv4 DHCP Class
3. Configure the IPv6 DHCP Profile
4. Configure the IPv6 DHCP Class

Configuring the IPv4 DHCP Server Profile

Use the following commands to configure the IPv4 DHCP server profile.

```
config
  profile dhcp dhcp_profile_name
  ipv4
  server { boot-filename boot_filename  } |
  { dns-servers dns_server } | { domain-name domain_name } |
  { netbios-name-server netbios_name_server } |
  { netbios-node-type { broadcast-node | hexadecimal | hybrid-node |
mixed-node | peer-to-peer-node } |
```

```
  { next-server ipv4_address } | { ntp-servers ipv4_address } |
  {{ pool-group-tag  pool_group_tag_name} | { pool-name ipam_pool_name }} |
  {option-codes option_codes_range { ascii-string value | force insert { true
 | false } | hex-string value |
  {ip-address ip_address } | { lease { days value | hours value| minutes value
 }
  exit
exit
```

NOTES:

- **profile dhcp** *dhcp_profile_name*: Specifies the DHCP profile name.
- **ipv4**: Enters IPv4 configuration mode.
- **server { boot-filename** *boot_filename* **} | { dns-servers** *dns_server* **} | { domain-name** *domain_name* **} | { netbios-name-server** *netbios_name_server* **} | { netbios-node-type { broadcast-node | hexadecimal | hybrid-node | mixed-node | peer-to-peer-node } | { next-server** *ipv4_address* **} | { ntp-servers** *ipv4_address* **} | { pool-name** *ipam_pool_name* **} | { option-codes** *option_codes_range* **{ ascii-string** *value* **| force insert { true | false } | hex-string** *value* **| { ip-address** *ip_address* **} | { lease { days** *value* **| hours** *value*| **minutes** *value* **}**: Specifies the IPv4 server details.
  - **boot-filename** *boot_filename*: Configures the boot file.
  - **dns-servers** *dns_server*: Specifies the Domain Name System (DNS) IPv4 servers available to a DHCP for an IPv4 client.
  - **domain-name** *domain_name*: Specifies the domain name for the IPv4 client.
  - **netbios-name-server** *netbios_name_server*: Configures the NetBIOS name servers.
  - **netbios-node-type { broadcast-node | hexadecimal | hybrid-node | mixed-node | peer-to-peer-node }**: Configures the NetBIOS node as a broadcast, hexadecimal, hybrid, mixed, or peer-to-peer node. The valid values for each of these nodes are:
    - **broadcast-node**: 0x1 B-node
    - **hexadecimal**: Operator provided custom 1 byte hex value
    - **hybrid-node**: 0x8 H-node
    - **mixed-node**: 0x4 M-node
    - **peer-to-peer-node**: 0x2 P-node
  - **next-server** *ipv4_address*: Specifies the TFTP-server IP address for the client to use.
  - **pool-name** *ipam_pool_name*: Specifies the IP Address Management (IPAM) assigned pool name.
  - **pool-group-tag** *pool_group_tag_name*: Specifies the group-tag mapping to IPAM pool configuration. Configure either **pool-name** or **pool-group-tag**, but not both.

    Specifies the tag group value of the pool. All pools carrying the same tag group value can be associated to a DHCP profile using the keyword **pool-group-tag** and the corresponding tag group value. The value must be a string.

- **option-codes option_codes_range { ascii-string** *value* **| force insert { true | false } | hex-string** *value* **| ip-address** *ip_address* **}**: Specifies the values for the ASCII string of length 128, force insert, hex string of length 128, or IP address (IPv4 IP address).
- **lease { days** *value* **| hours** *value* **| minutes** *value* **}**: Specifies the lease time duration in the number of days, hours, and minutes. The number of lease days supported is from 0 to 365. The number of leave hours supported ranges from 0 to 23 and minutes from 0 to 59.

Configuring the IPv4 DHCP Class

Use the following commands to configure the IPv4 DHCP class.

```
config
  profile dhcp dhcp_profile_name
  ipv4
  class dhcp_class_name
  matches { match { dhcpv4-circuit-id { ascii value | hex value } |
dhcpv4-remote-id { ascii value |
    hex value } | dhcpv4-vendor-class { ascii value | hex value } |
dhcpv4-user-class { ascii value |
    hex value } } | match-type { all match_key_value | any match_key_value } }
  server { boot-filename boot_filename } |
     { dns-servers dns_server } | { domain-name domain_name } |
     { netbios-name-server netbios_name_server } |
     { netbios-node-type { broadcast-node | hexadecimal | hybrid-node |
mixed-node | peer-to-peer-node } |
     { next-server ipv4_address } | { ntp-servers ipv4_address } |
     {{ pool-group-tag  pool_group_tag_name} | { pool-name ipam_pool_name }} |
    {option-codes option_codes_range { ascii-string value | force insert {
true | false } | hex-string value |
    {ip-address ip_address } | { lease { days value | hours value| minutes
value }
  end
```

NOTES:

- **profile dhcp** *dhcp_profile_name*: Specifies the DHCP profile name.
- **ipv4**: Enters IPv4 configuration mode.
- **class** *dhcp_class_name*: Creates a proxy profile class (DHCP), which can be used to enter the proxy profile class sub-configuration mode.
- **matches { match { dhcpv4-circuit-id { ascii** *value* **| hex** *value* **} | dhcpv4-remote-id { ascii** *value* **| hex** *value* **} | dhcpv4-vendor-class { ascii** *value* **| hex** *value* **} | dhcpv4-user-class { ascii** *value* **| hex** *value* **} } | match-type { all** *match_key_value* **| any** *match_key_value* **} }**: Specifies the list of match keys and values. The match values supported are DHCPv4 circuit ID, DHCPv4 remote ID, DHCPv4 vendor class, and DHCPv4 user class. Each of the values must specify either an ASCII or hexadecimal value.

  match-type { all | any }: Specifies if the match value should apply to any of the specified keys or to all the keys.
- **server { boot-filename** *boot_filename* **} | { dns-servers** *dns_server* **} | { domain-name** *domain_name* **} | { netbios-name-server** *netbios_name_server* **} | { netbios-node-type { broadcast-node | hexadecimal**

| hybrid-node | mixed-node | peer-to-peer-node } | { next-server *ipv4_address* **} | { ntp-servers** *ipv4_address* **} | { pool-name** *ipam_pool_name* **} | { option-codes** *option_codes_range* **{ ascii-string** *value* **| force insert { true | false } | hex-string** *value* **| { ip-address** *ip_address* **} | { lease { days** *value* **| hours** *value*| **minutes** *value* **}**: Specifies the IPv4 server details.

- **boot-filename** *boot_filename*: Configures the boot file.
- **dns-servers** *dns_server*: Specifies the Domain Name System (DNS) IPv4 servers available to a DHCP for an IPv4 client.
- **domain-name** *domain_name*: Specifies the domain name for the IPv4 client.
- **netbios-name-server** *netbios_name_server*: Configures the NetBIOS name servers.
- **netbios-node-type { broadcast-node | hexadecimal | hybrid-node | mixed-node | peer-to-peer-node }**: Configures the NetBIOS node as a broadcast, hexadecimal, hybrid, mixed, or peer-to-peer node. The valid values for each of these nodes are:
  - **broadcast-node**: 0x1 B-node
  - **hexadecimal**: Operator provided custom 1 byte hex value
  - **hybrid-node**: 0x8 H-node
  - **mixed-node**: 0x4 M-node
  - **peer-to-peer-node**: 0x2 P-node
- **next-server** *ipv4_address*: Specifies the TFTP-server IP address for the client to use.
- **pool-name** *ipam_pool_name*: Specifies the IP Address Management (IPAM) assigned pool name.
- **pool-group-tag** *pool_group_tag_name*: Specifies the group-tag mapping to IPAM pool configuration. Configure either **pool-name** or **pool-group-tag**, but not both.

  Specifies the tag group value of the pool. All pools carrying the same tag group value can be associated to a DHCP profile using the keyword **pool-group-tag** and the corresponding tag group value. The value must be a string.
- **option-codes option_codes_range { ascii-string** *value* **| force insert { true | false } | hex-string** *value* **| ip-address** *ip_address* **}**: Specifies the values for the ASCII string of length 128, force insert, hex string of length 128, or IP address (IPv4 IP address).
- **lease { days** *value* **| hours** *value* **| minutes** *value* **}**: Specifies the lease time duration in the number of days, hours, and minutes. The number of lease days supported is from 0 to 365. The number of leave hours supported ranges from 0 to 23 and minutes from 0 to 59.

Configuring the IPv6 DHCP Server Profile

Use the following commands to configure the IPv6 DHCP server profile.

```
config
  profile dhcp dhcp_profile_name
    ipv6
      server { aftr-name value | dns-servers dns_server
      domain-name domain_name |
```

```
        {{ iana-pool-name ipam_pool_name}  | {iana-pool-group-tag
iana_pool_group_tag_name}}|
        {{iapd-pool-name ipam_pool_name } | {iapd-pool-group-tag
iapd_pool_group_tag_name}}|
        lease { days value | hours value | minutes value }|
        preference value }
```

NOTES:

- **profile dhcp** *dhcp_profile_name*: Specifies the DHCP profile name.
- **ipv6**: Enters IPv6 configuration mode.
- **server { aftr-name** *value* **| dns-servers** *dns_server* **| domain-name** *domain_name* **| iana-pool-name** *ipam_pool_name* **| iapd-pool-name** *ipam_pool_name*| **lease { days** *value* **| hours** *value* **| minutes** *value* **} | preference** *value* **}**: Specifies the IPv6 server details.
  - **aftr-name** *value*: Specifies the FQDN string.
  - **dns-servers** *dns_server*: Specifies the Domain Name System (DNS) IPv4 servers available to a DHCP for an IPv4 client.
  - **domain-name** *domain_name*: Specifies the domain name for the IPv4 client.
  - **iana-pool-name** *ipam_pool_name*: Specifies the Internet Assigned Numbers Authority (IANA) pool name.
  - **iana-pool-group-tag** *iana_pool_group_tag_name*: Specifies the tag group value of the pool. All pools carrying the same tag group value can be associated to a DHCP profile using the keyword **pool-group-tag** and the corresponding tag group value. The value must be a string. Configure either **iana-pool-name** or **iana-pool-group-tag** , but not both.
  - **iapd-pool-name** *ipam_pool_name*: Specifies the Identity Association for Prefix Delegation (IAPD) pool name.
  - **iapd-pool-group-tag** *iapd_pool_group_tag_name*: Specifies the tag group value of the pool. All pools carrying the same tag group value can be associated to a DHCP profile using the keyword **pool-group-tag** and the corresponding tag group value. The value must be a string. Configure either **iapd-pool-name** or **iapd-pool-group-tag** , but not both.
  - **lease { days** *value* **| hours** *value* **| minutes** *value* **}**: Specifies the lease time duration in the number of days, hours, and minutes. The number of lease days supported is from 0 to 365. The number of leave hours supported ranges from 0 to 23 and minutes from 0 to 59.
  - **preference** *value*: Specifies the DHCP server preference. The preference value ranges from 1 to 255.

Configuring the IPv6 DHCP Class

Use the following commands to configure the IPv6 DHCP class.

```
config
  profile dhcp dhcp_profile_name
    ipv6
      class dhcp_class_name
```

```
        server { aftr-name value | dns-servers dns_server | domain-name
domain_name |
        {{iana-pool-name ipam_pool_name  } | {iana-pool-group-tag
pool_group_tag_name}} |
        {{iapd-pool-name ipam_pool_name } | {iapd-pool-group-tag
pool_group_tag_name}} |
        lease { days value |
        hours value | minutes value }
        preference value
        end
```

NOTES:

- **profile dhcp** *dhcp_profile_name*: Specifies the DHCP profile name.
- **ipv6**: Enters IPv6 configuration mode.
- **class** *dhcp_class_name*: Creates a proxy profile class (DHCP), which can be used to enter the proxy profile class sub-configuration mode.
- **server { aftr-name** *value* **| dns-servers** *dns_server* **| domain-name** *domain_name* **| iana-pool-name** *ipam_pool_name* **| iapd-pool-name** *ipam_pool_name* **| lease { days** *value* **| hours** *value* **| minutes** *value* **} | preference** *value* **}**: Specifies the IPv6 class server details.
  - **aftr-name** *value*: Specifies the FQDN string.
  - **dns-servers** *dns_server*: Specifies the Domain Name System (DNS) IPv6 servers available to a DHCP for an IPv6client.
  - **domain-name** *domain_name*: Specifies the domain name for the IPv6 client.
  - **iana-pool-name** *ipam_pool_name*: Specifies the Internet Assigned Numbers Authority (IANA) pool name.
  - **iana-pool-group-tag** *iana_pool_group_tag_name*: Specifies the tag group value of the pool. All pools carrying the same tag group value can be associated to a DHCP profile using the keyword **pool-group-tag** and the corresponding tag group value. The value must be a string. Configure either **iana-pool-name** or **iana-pool-group-tag**, but not both.
  - **iapd-pool-name** *ipam_pool_name*: Specifies the Identity Association for Prefix Delegation (IAPD) pool name.
  - **iapd-pool-group-tag** *iapd_pool_group_tag_name*: Specifies the tag group value of the pool. All pools carrying the same tag group value can be associated to a DHCP profile using the keyword **pool-group-tag** and the corresponding tag group value. The value must be a string. Configure either **iapd-pool-name** or **iapd-pool-group-tag** , but not both.
  - **lease { days** *value* **| hours** *value* **| minutes** *value* **}**: Specifies the lease time duration in the number of days, hours, and minutes. The number of lease days supported is from 0 to 365. The number of leave hours supported ranges from 0 to 23 and minutes from 0 to 59.
  - **preference** *value*: Specifies the DHCP server preference. The preference value ranges from 1 to 255.

DHCPv6 Raw Option Support

Table 5: Feature History

Feature Name	Release Information	Description
DHCPv6 Raw Option Support	2024.04.0	cnBNG now supports DHCPv6 raw options, enabling you to set any DHCPv6 option type, including Mapping of Address and Port using Encapsulation (MAP-E) for scenarios such as migration to cloud native BNG.

This feature introduces the capability to configure DHCPv6 options in their raw form. This means you can specify any DHCPv6 option type beyond the predefined set, providing greater flexibility in host configurations.

DHCPv6 Raw Option Support is critical for scenarios such as migration from physical BNG to cloud native BNG, where specific DHCPv6 options like MAP-E (DHCPv6 option 94) need to be set for MAP-E translation.

The raw option support is designed to integrate smoothly with existing DHCP server configurations. You can continue to use the familiar settings for common attributes (for example, DNS, domain name, lease time) while leveraging the raw option feature for more specialized requirements, and different network environments.

Configure DHCPv6 Raw Option Support

You can configure option codes in both DHCP profile and DHCP class sub mode. As with existing configurations, priority is given to the class configuration.

Procedure

Step 1 Configure the option codes in IPv6 DHCP server profile.

Example:

```
config
  profile dhcp dhcp_profile_name
   ipv6
     mode server
     server
      option-codes
      option-code 39
       ascii-string abcd.com
     exit
     option-code 31
      ip-address [ 2001:db8:c:641::6401 ]
    exit
```

The following is a sample MAP-E configuration.

```
profile dhcp DHCP_PROFILE
ipv6
```

```
  mode server
  server
   option-codes
    option-code 39
     hex-string
00590016000c1a5e4102002c2a020586f900005d0004040000000059001500 1018c63365002820010db896005d0004040000000005a00102a020586000000000000000000000406

    exit
   exit
  exit
exit
exit
```

Step 2 Configure the option codes in IPv6 DHCP class.

Example:

```
config
   profile dhcp dhcp_profile_name
     ipv6
        class dhcp_class_name
        server
           option-codes
            option-code 23
             ip-address [ 2001::7 ]
          option-code 39
             ascii-string class.com
    exit

....
```

NOTES:

- **option-codes**: Enters the Option Codes Configuration mode (config-option-codes). Configures the OptionCode table.
- **option-code** *code*: Specifies the DHCP option code [up to 255].
- **ascii-string** *value*: Specifies the values for the ASCII string of length 256.
- **ip-address** *ip_address* : Specifies the IPv6 address.
- **force-insert** {*true* | *false* }: If set to true, force insert the option regardless of the DHCPv6 Option Request Option (ORO) value. If set to false, honor the ORO option.

The DHCP class name (for example, `automation-class`) is applied to a subscriber if the user profile has the DHCP class attribute `Cisco-AVPair += "dhcp-class=automation-class"` set. The following are a few scenarios based on the configuration example:

- **Option Code 23**
  - **Subscriber A:** For whom `dhcp-class=automation-class` is set in the user profile, option 23 is included in response packets with the IP address `[2001::7]`.
  - **Subscriber B:** For whom `dhcp-class` is not set in the user profile, option 23 will not be included in response packets.

- **Option Code 39**
  - **Subscriber A:** For whom `dhcp-class=automation-class` is set in the user profile, option 39 is included in response packets with the value `"class.com"`.
  - **Subscriber B:** For whom `dhcp-class` is not set in the user profile, option 39 will be included in response packets with the value `"abcd.com"`.
- **Option Code 31**
  - **Subscriber A:** For whom `dhcp-class=automation-class` is set in the user profile, option 31 would be included in response packets with the IP address `2001:db8:c:641::6401`.
  - **Subscriber B:** For whom `dhcp-class` is not set in the user profile, option 31 will still be included in response packets with the IP address `"2001:db8:c:641::6401"`

Raw Option Code Exclusion List

The following list of options cannot be set using the raw option CLIs. These options are either not intended to be set by the server or require additional functionalities beyond simple option setting:

```
OPTION_CLIENTID - 1
OPTION_SERVERID - 2
OPTION_IA_NA - 3
OPTION_IA_TA - 4
OPTION_IAADDR - 5
OPTION_ORO - 6
OPTION_PREFERENCE - 7
OPTION_ELAPSED_TIME - 8
OPTION_RELAY_MSG - 9
Option - 10
OPTION_STATUS_CODE - 13
OPTION_RAPID_COMMIT - 14
OPTION_USER_CLASS - 15
OPTION_INTERFACE_ID - 18
OPTION_DNS_SERVERS -23
OPTION_DOMAIN_LIST - 24
OPTION_IA_PD - 25
OPTION_IAPREFIX -26
OPTION_INFORMATION_REFRESH_TIME - 32
Option - 35
OPTION_REMOTE_ID - 37
OPTION_SUBSCRIBER_ID - 38
OPTION_CLIENT_FQDN - 39
OPTION_ERO - 43
OPTION_LQ_QUERY - 44
OPTION_LQ_QUERY - 45
OPTION_CLT_TIME - 46
OPTION_LQ_RELAY_DATA - 47
OPTION_LQ_CLIENT_LINK - 48
OPTION_RELAY_ID - 53
OPTION_AFTR_NAME - 64
OPTION_RSOO - 66
OPTION_PD_EXCLUDE - 67
OPTION_VSS - 68
OPTION_CLIENT_LINKLAYER_ADDR - 79
OPTION_LINK_ADDRESS - 80
OPTION_RADIUS - 81
OPTION_DHCPV4_MSG - 87
OPTION_DHCP4_O_DHCP6_SERVER - 88
OPTION_LQ_BASE_TIME - 100
OPTION_LQ_START_TIME - 101
```

```
OPTION_LQ_END_TIME - 102
OPTION_ANI_ATT - 105
OPTION_ANI_NETWORK_NAME - 106
OPTION_ANI_AP_NAME - 107
OPTION_ANI_AP_BSSID - 108
OPTION_ANI_OPERATOR_ID - 109
OPTION_ANI_OPERATOR_REALM - 110
OPTION_MUD_URL_V6 - 112
OPTION_F_BINDING_STATUS - 114
OPTION_F_CONNECT_FLAGS - 115
OPTION_F_DNS_REMOVAL_INFO - 116
OPTION_F_DNS_HOST_NAME - 117
OPTION_F_DNS_ZONE_NAME - 118
OPTION_F_DNS_FLAGS - 119
OPTION_F_EXPIRATION_TIME - 120
OPTION_F_MAX_UNACKED_BNDUPD - 121
OPTION_F_MCLT - 122
OPTION_F_PARTNER_LIFETIME - 123
OPTION_F_PARTNER_LIFETIME_SENT - 124
OPTION_F_PARTNER_DOWN_TIME - 125
OPTION_F_PARTNER_RAW_CLT_TIME - 126
OPTION_F_PROTOCOL_VERSION - 127
OPTION_F_KEEPALIVE_TIME - 128
OPTION_F_RECONFIGURE_DATA - 129
OPTION_F_RELATIONSHIP_NAME - 130
OPTION_F_SERVER_FLAGS - 131
OPTION_F_SERVER_STATE - 132
OPTION_F_START_TIME_OF_STATE - 133
OPTION_F_STATE_EXPIRATION_TIME - 134
OPTION_RELAY_PORT - 135
OPTION_IA_LL - 138
OPTION_LLADDR - 139
OPTION_SLAP_QUAD – 140
```



Note

- The CLI supports setting option codes up to 255, excluding those specified in the **Raw Option Code Exclusion List**.
- The **force insert** flag setting for a given option code must be consistent across the configuration. For example, if a given option code is configured under a class with the **force insert** flag set to **TRUE**, it must also be set to **TRUE** if the same option code is configured under a profile, and conversely.

IPv6 Class Configuration and Static IP Allocation Support

Table 6: Feature History

Feature Name	Release Information	Description
IPv6 Class Configuration and Static IP Allocation Support	2025.01.0	This feature ensures reliable IPv6 client IP allocation by allowing static IP assignments based on class parameters within a DHCPv6 profile.

This feature allows static IP address allocation for clients based on their classification within the DHCPv6 profile. This ensures that specific devices receive predetermined IP addresses based on their identity or other

classification criteria. The feature involves configuring a DHCPv6 profile that contains DHCP options for IPv6. Within this profile, classes can be defined that include specific DHCP options and criteria for selecting which class a client belongs to.

Class selection based on Match-Info:

You can configure the class selection to be based on a match of **any** or **all** key parameters of that class. If no DHCP option class matches for the ongoing session or any requested DHCP Options is not found in the selected class, then the requested options are selected from the default DHCP options of that profile.

Key parameters for DHCPv6 Class selection:

The key parameters to select the DHCP Options Class for DHCPv6 include:

- Interface-id (DHCP Option 18)
- Remote-id (DHCP Option 37)
- Vendor-class (DHCP Option 16)
- User-class (DHCP Option 15)

The DHCPv6 Options Class consists of the following elements:

- IanaPoolName
- IapdPoolName
- DnsServers
- DomainName
- Preference
- AftrName
- Lease
- Static-ip-key

Behavior of Class Selection

- When multiple classes are configured, if the incoming solicit matches more than one class, the order of matching is not specified. The solicit may match any one of the classes.

  For example, if an incoming packet has options that match both class1 and class2, it can match either class, and the matching order may differ from the order in the running configuration.

- If the packet doesn’t match any of the configured classes, it gets an IP from the default pool.

Static IP Key Identifier

It is expected that there is an IP configured in the IPv6 static database for the corresponding MAC address. If no IP address is configured for the matching MAC address, IP allocation fails.

Match-Type Scenarios

- **Match-Type "All":**

With **match-type all**, a solicit must match all configured option values to match the class.

- **Match-Type "Any":**

  With **match-type any**, the class is selected if any one of the specified key parameters matches. This means that the incoming packet only needs to satisfy at least one of the configured match criteria (for example, interface-id, remote-id, vendor-class, or user-class) for the class to be considered a match.

Priority of DHCP-Class Attribute

If the **dhcp-class** attribute from RADIUS and match type coexist, the RADIUS configuration takes precedence.

Configure the IPv6 DHCP Class

Procedure

Define a DHCP profile with IPv6 configuration:

Example:

```
config
  profile dhcp dhcp_profile_name
  ipv6
    class dhcp_class_name

     matches
      match-type { any  match_key_value | all match_key_value }
      match dhcpv6-interface-id { ascii value | hex value }
      match dhcpv6-remote-id { ascii value | hex value }
      match dhcpv6-user-class { ascii value | hex value }
      match dhcpv6-vendor-class { ascii value | hex value }
     exit
    exit
```

The following is a sample configuration for match-type **any**:

```
profile dhcp dhcp-profile1
Ipv6
 mode server
  server
   iana-pool-name poolv6iana
   iapd-pool-name poolv6iapd
   dns-servers    [ 5000::5 ]
   domain-name    cisco.com
   lease days 1
exit
 class class1
   matches
    match-type any
    match dhcpv6-interface-id hex [ 61626365 ]
    match dhcpv6-remote-id hex [ 6656 ]
    match dhcpv6-user-class hex [ 4a696f53746174696332 ]
    match dhcpv6-vendor-class hex [ 726f75746572646576696532 ]
exit
```

Example Scenarios for Match-type any

- If the incoming packet includes a remote-id with the hex value 6656, an interface-id with the hex value 567644, a user-class with the hex value 6a87d556, and a vendor-class with the hex value 678776346, the class matches. This is because at least one of the specified conditions is met.
- If the incoming packet has a remote-id with the hex value 34566, an interface-id with the hex value 245644, a user-class with the hex value 86787d556, and a vendor-class with the hex value 776d346, the class does not match. This is because none of the specified conditions are met.

The following is a sample configuration for match-type **all**:

```
profile dhcp dhcp-profile1
Ipv6
 mode server
  server
   iana-pool-name poolv6iana
   iapd-pool-name poolv6iapd
   dns-servers    [ 5000::5 ]
   domain-name    cisco.com
   lease days 1
exit
 class class1
   matches
    match-type all
    match dhcpv6-interface-id hex [ 61626365 ]
    match dhcpv6-remote-id hex [ 6656 ]
    match dhcpv6-user-class hex [ 4a696f53746174696332 ]
    match dhcpv6-vendor-class hex [ 726f7574657264657669636532 ]
exit
```

Example Scenarios for Match-type all

- If the incoming packet contains an interface-id with the hex value 61626365, a remote-id with the hex value 2356, a user-class with the hex value 6a87d556, and a vendor-class with the hex value 678776346, the class does not match. This means not all specified conditions are met.
- If the incoming packet has an interface-id with the hex value 61626365, a remote-id with the hex value 6656, a user-class with the hex value 4a696f53746174696332, and a vendor-class with the hex value 726f7574657264657669636532, the class matches. This indicates that all the specified conditions are satisfied.

Note Each match option can have multiple values. For example,

```
match dhcpv6-interface-id hex [ 61626367 ]

match dhcpv6-user-class hex [ 4a696f53746174696111 4a696f53746174696112 4a696f53746174696113
 ]
```

NOTES:

- **profile dhcp** *dhcp_profile_name*: Specifies the DHCP profile name.
- **ipv6**: Enters IPv6 configuration mode.
- **class** *dhcp_class_name*: Creates a proxy profile class (DHCP), which can be used to enter the proxy profile class sub-configuration mode.

- **matches { match { dhcpv6-interface-id { ascii** *value* **| hex** *value* **} | dhcpv6-remote-id { ascii** *value* **| hex** *value* **} | dhcpv6-vendor-class { ascii** *value* **| hex** *value* **} | dhcpv6-user-class { ascii** *value* **| hex** *value* **} }** : Specifies the list of match keys and values. Each of the values must specify either an ASCII or hexadecimal value.

  match-type { all | any }: Specifies if the match value should apply to any of the specified keys or to all the keys.

Configure Static IP Allocation

Procedure

Step 1 Configure the static IP key.

Example:

```
config
  profile dhcp dhcp_profile_name
 ipv6
  class dhcp_class_name
   server
    iana-pool-name iana_pool_name
    iapd-pool-name iapd_pool_name
    static-ip-key
     identifier client-mac-address mac_address
    exit
   exit
 exit
```

NOTES:

- **static-ip-key identifier client-mac-address** *mac_address*: Enables a client with a specific MAC address to receive a pre-configured static IP address from the IPv6 static database.

  Note For the allocation to succeed, there must be an IP address already configured in the IPv6 static database corresponding to the client's MAC address. If no such IP address is configured, the IP allocation will fail.

Step 2 Configure static IP address ranges.

Example:

```
config
  ipam
    instance instance_id
      address-pool pool_name
       vrf-name vrf_name
       static enable user_plane_name
       ipv4
```

```
        split-size
         no-split
        exit
        address-range start_ipv4_address end_ipv4_address
          default-gateway ipv4_address
          exit
      ipv6
        address-ranges
          split-size
            no-split
          exit
         address-range start_ipv6_address end_ipv6_address
        exit
        prefix-ranges
          split-size
           no-split
          exit
          prefix-range ipv6_address length prefix_length
        exit
    exit
```

The following is a sample configuration:

```
ipam
 instance 1
  address-pool staticpool1
   vrf-name vrf1
   static enable user-plane1
   ipv4
    split-size
     no-split
    exit
    address-range 10.44.1.0 10.44.1.255 default-gateway 10.44.1.1
   exit
   ipv6
    address-ranges
     split-size
      no-split
     exit
     address-range 2001:db8::1 2001:db8::ffff
    exit
    prefix-ranges
     split-size
      no-split
     exit
     prefix-range 2002:ab:: length 48
    exit
   exit
```

NOTES:

- **static enable** *user_plane_name*: Configures static IP details. Sets the specified User Plane (UP) as static. *user_plane_name* is the specified UP for this static pool.
- **split-size no-split**: Specifies that the address-ranges should not be into smaller chunks.
- **address-range** *start_ipv4/ipv6_address end_ipv4/ipv6_address*: Configures the IPv4 or IPv6 address range with the starting and ending IPv4/IPv6 address.

- IPAM does only route-validation for Static IP.

Leased IP Hold Time

Feature Summary and Revision History

Summary Data

Table 7: Summary Data

Applicable Product(s) or Functional Area	cnBNG
Applicable Platform(s)	SMI
Feature Default Setting	Disabled - Configuration Required
Related Documentation	Not Applicable

Revision History

Table 8: Revision History

Revision Details	Release
First introduced.	2022.02.0

Feature Description

The Leased IP Hold Time feature provides the flexibility to configure the cnBNG DHCP server to apply a hold time to the leased IP. After release from the client or lease expiry, the IP is not available for reuse for other clients until the hold time elapses.

After the lease expiry or release from client, if the same client requests the IP again within the hold time, the same IP is provided to the client. After the hold time, the IP is made available in the free IP list and can get allocated to other clients.

A new CLI is introduced in the DHCP profile to allow the hold time configuration. For more information, see Configuring Leased IP Hold Time, on page 30.

How it Works

This section provides a brief of how the Leased IP Hold Time feature works.

The DHCP record retains minimal data (like, sublabel, IP, pool information, and so on) in the CDL, to hold the IP for a given session after release or timer expiry after a subscriber is disconnected.

The FSM state moves back to the initialization state for the IP. However, the IP is not be released back to the pool (basically no interaction with IPAM or Node Manager). The information for the session will be unavailable in the Session Manager and User Plane function (UPF). The DHCP record with the IP on hold is removed after the hold time if the session does not come back up within the specified hold time.

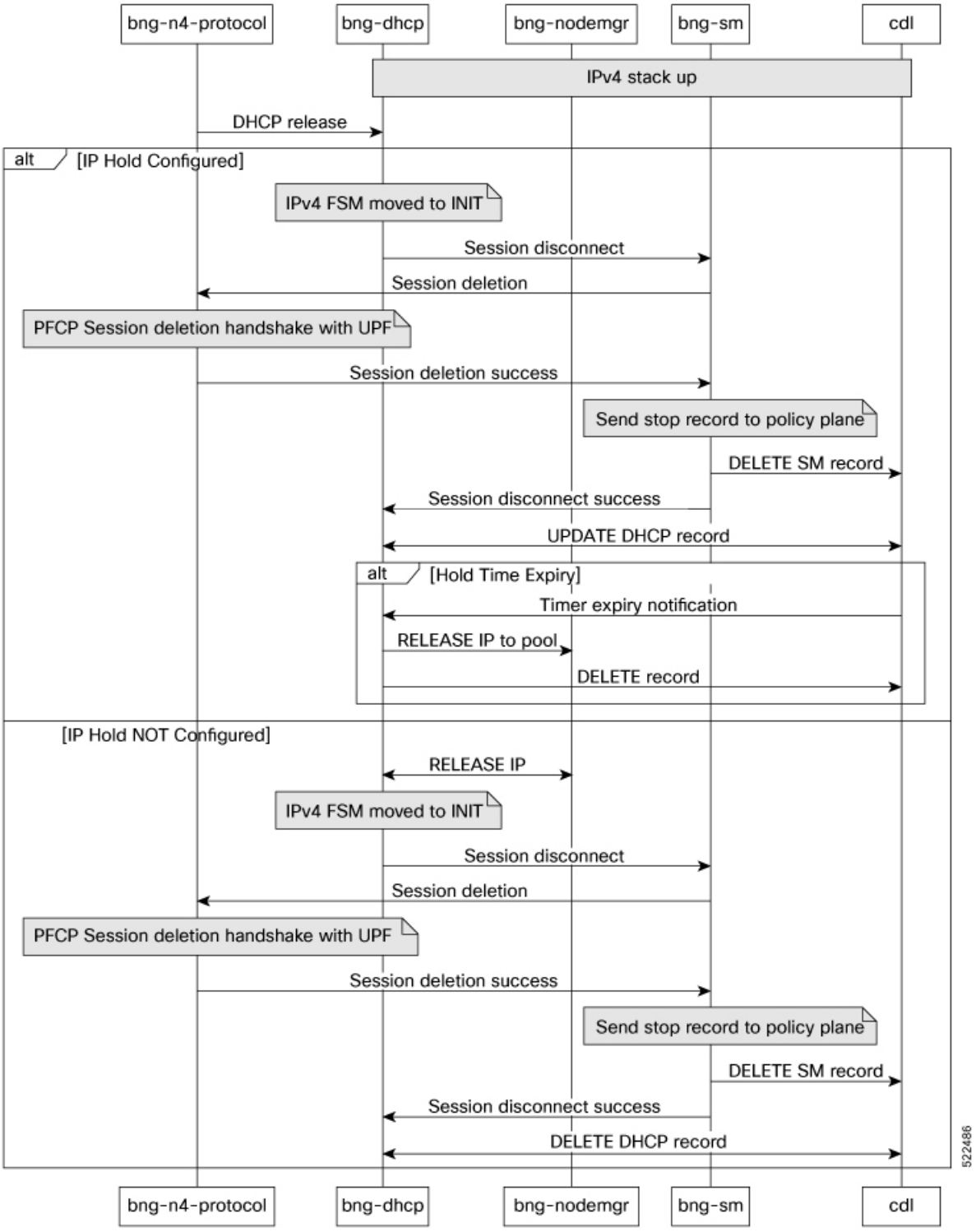
Call Flows

This section includes the following call flows.

IPv4 Session with or without Hold Time Call Flow

The following is an example of an IPv4 release for an IPv4 session with and without a hold time applied.

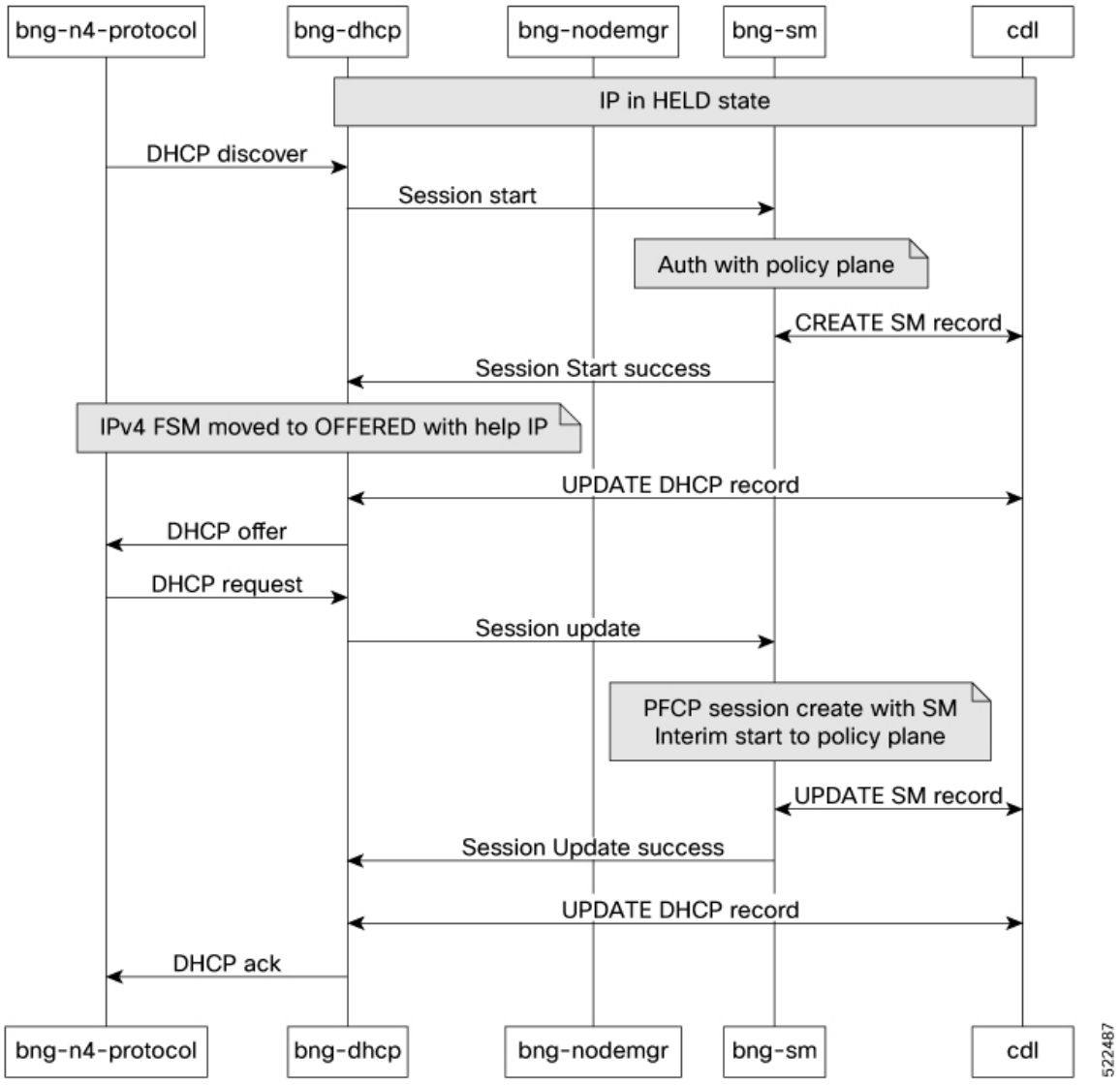
Figure 2: IPv4 Session with or without Hold Time Call Flow



IPv4 Session Bring-Up with IP on Hold

The following is an example of an IPv4 serssion bring-up with IP on hold.

Figure 3: IPv4 Session Bring-Up with IP on Hold Call Flow



Limitations and Restrictions

The Leased IP Hold Time feature has the following limitations:

- When the administrator runs the **clear** and **coa disconnect** CLI comamnds, it cleans up the session even if the IP is in 'hold' state. After the administrato clears the session, the hold timer for a new IP does not take effect.
- If the IP is declined from the client side, the IP is not moved to 'hold' state.
- When Request, Renew, Rebind are not acknowledged due to internal errors or session establishment or modification failures, the session IP is not maintained in the 'hold' state.

- Class level hold time configuration is not supported.
- Only IPoE sessions are qualified for this feature.

Configuring the Leased IP Hold Time Feature

This section describes how to configure the Leased IP Hold Time feature.

Configuring the Leased IP Hold Time feature involves the following procedure.

Configuring Leased IP Hold Time

Use the following commands to configure the leased IP hold time on the DHCP server profile.

```
config
  profile dhcp dhcp_profile_name
  { ipv4 | ipv6 }
  server hold-time minutes timer_interval
  exit
```

NOTES:

- **profile dhcp** *dhcp_profile_name*: Specifies the DHCP profile name.
- **{ ipv4 | ipv6 }**: Enters IPv4 or IPv6 configuration mode.
- **server hold-time minutes** *timer_interval*: Specifies the leased IP hold time in minutes for the IPv4 or IPv6 server. *timer_interval* must be an integer in the range of 1 to 59 minutes.

DHCP IP Lease Reservation

Feature Summary

Table 9: Feature Summary

Applicable Product(s) or Functional Area	cnBNG
Applicable Platform(s)	SMI
Feature Default Setting	Disabled - Configuration Required
Related Documentation	Not Applicable

Revision History

Table 10: Revision History

Revision Details	Release
First introduced	2022.04.0

Feature Description

DHCP IP Lease Reservation feature enables the DHCP to allocate an IP address dynamically when the subscriber logs into the network the first time. Then, the assigned IP address can be reserved permanently for the subscriber, which means, the same IP address is assigned every time the subscriber logs in.

How it Works

This section provides a brief of how the DHCP IP Lease Reservation feature works.

After the DHCP IP Lease Reservation feature is enabled (see Configuring DHCP IP Lease Reservation, on page 31), if a subscriber (CPE) logs into the system for the first time, IPAM allocates an IP address dynamically from the IP pool. Administrators can use the REST API/action command (see Reserving IP Address using CLI (Action Command/REST API), on page 31) to reserve the IP address for the subscriber. So, when the same session is initiated the next time, the DHCP provides the same IP address to the subscriber.



Note If you do not want to reserve the IP address, the administrators can use the same REST API/action command with **delete** option and clear the IP lease reservation.

Limitations and Restrictions

The DHCP IP Lease Reservation feature has the following limitation:

- The DHCP IP Lease Reservation and Leased IP Hold Time features cannot be used together at the same time.

Configuring DHCP IP Lease Reservation

Use the following commands to enable/disable the DHCP IP Lease Reservation feature:

```
config
  [ no ] subscriber featurette dhcp-lease-reservation enable
end
```

NOTES:

- **subscriber featurette dhcp-lease-reservation enable**: Enables the DHCP IP Lease Reservation feature
- **no subscriber featurette dhcp-lease-reservation enable**: Disables the DHCP IP Lease Reservation feature

Reserving IP Address using CLI (Action Command/REST API)

Administrators can use the following action command/REST API to reserve the addresses (IPv4, IANA, and IAPD) that are allocated to the subscriber with a specific username.

```
bng# subscriber lease-reservation subkey username_string [ delete ]
```

NOTES:

- **subkey** *username_string* : Specifies the username for which the IP addresses are reserved.
- **delete**: Clears the lease reservation for the specific username.

Note

- This command/REST API fails if the subscriber is disconnected.
- This command/REST API fails if the DHCP IP Lease Reservation feature is not enabled.

Enhanced support for RADIUS attributes

We have enhanced the RADIUS attributes, especially Stateful-IPv6-Address-Pool, and Delegated-IPv6-Prefix-Pool. These enhancements are crucial for effectively managing and optimizing your modern network infrastructure.

Stateful-IPv6-Address-Pool support

Stateful-IPv6-Address-Pool is a RADIUS attribute that

- enables the RADIUS server to send the name of an assigned pool for a subscriber to the cnBNG Control Plane in Access-Accept, which uses it to assign IPv6 addresses to subscribers via the DHCPv6 protocol
- overrides any local IPv6 address pool configured in the cnBNG CP DHCPv6 profile, and
- is defined in RFC 6911 - *RADIUS Attributes for IPv6 Access Networks.*

Table 11: Feature History

Feature Name	Release Information	Description
Stateful-IPv6-Address-Pool support	2025.02.0	We have enhanced IPv6 address allocation by enabling the RADIUS server to specify the assigned pool name to cnBNG. The cnBNG uses this information to assign IPv6 addresses to subscribers.

IPv6 address allocation

This attribute specifies the name of a pool that must be used by the cnBNG to select an IPv6 address for a user.

Stateful-IPv6-Address-Pool fields

Attribute	Type	Length	String
Stateful-IPv6-Address-Pool	172	Minimum length in bytes: 3.	Contains the name of an assigned IPv6 stateful address pool configured on the cnBNG. This field is not terminated by NULL (hexadecimal 00).

Configure Stateful-IPv6-Address-Pool attribute

You can configure the **Stateful-IPv6-Address-Pool** attribute using the user-profile on the RADIUS server. You can then directly receive it from the RADIUS server. No configuration is needed on the cnBNG CP side.

Procedure

Step 1 Define the **Stateful-IPv6-Address-Pool** attribute on the RADIUS server.

Example:

```
user@example.com    Password="abc"
    Service-Type=Framed-User,
    Stateful-IPv6-Address-Pool="dhcpv6-iana-pool"
```

Step 2 Use the **show subscriber session detail** command to verify the configuration.

Example:

```
bng# show subscriber session detail
Mon Apr  7  10:05:13.666 UTC+00:00
subscriber-details
{
  "subResponses": [
    {
      "subLabel": "16777219",
      "mac": "cc11.0000.0001",
      "acct-sess-id": "01000003",
      "upf": "asr9k-1",
      "port-id": "Bundle-Ether1.100",
      "up-subs-id": "3",
      "sesstype": "ppp",
      "state": "established",
      "subCreateTime": "Mon, 07 Apr 2025 10:04:53 UTC",
      "subsAttr": {
        "attrs": {
          "Authentic": "RADIUS(1)",
          "Framed-Protocol": "PPP(1)",
          "addr": "33.0.0.4",
          "addrv6": "2:3::202",
          "circuit-id-tag": "circuit1",
          "client-mac-address": "cc11.0000.0001",
          "connect-progress": "IPV6CP Open",
          "delegated-prefix": "3002:db0:0:1::/64",
          "dhcpv6-client-id": "0x00030001cc1100000001",
          "inner-vlan-id": "200",
          "outer-vlan-id": "100",
        }
      },
      "subcfgInfo": {
        "committedAttrs": {
          "attrs": {
            "Stateful-IPv6-Address-Pool": "dhcpv6-iana-pool",
            "accounting-list": "aaa-prof1",
            "acct-interval": "2000",
            "addr-pool": "ipv4-pool",
            "delegated-ipv6-pool": "dhcpv6-iapd-pool",
```

```
"session-acct-enabled": "true",
"vrf": "default"
```

Delegated-IPv6-Prefix-Pool support

Delegated-IPv6-Prefix-Pool is an attribute that

- allows a RADIUS server to send the name of a prefix pool for a subscriber to the cnBNG Control Plane in Access-Accept, which uses it to assign IPv6 prefixes to subscribers via the DHCPv6 protocol
- overrides any local IPv6 prefix pool configured in the cnBNG CP DHCPv6 profile, and
- is defined in RFC 6911 - *RADIUS Attributes for IPv6 Access Networks.*

Table 12: Feature History

Feature Name	Release Information	Description
Delegated-IPv6-Prefix-Pool support	2025.02.0	We have enhanced IPv6 prefix delegation by allowing the RADIUS server to specify a prefix pool name to cnBNG. The cnBNG uses this information to assign IPv6 prefixes to subscribers.

IPv6 prefix delegation

This attribute specifies the name of a pool that should be used by the cnBNG to select an IPv6 delegated prefix for a user.

Useful in dynamic IPv6 environments

This attribute is useful in environments where dynamic assignment of IPv6 prefixes is necessary, and it helps streamline the process of prefix delegation by providing predefined pool names that guide the selection process.

Delegated-IPv6-Prefix-Pool attribute fields

Attribute	Type	Length	String
Delegated-IPv6-Prefix-Pool	171	Minimum length in bytes: 3.	Contains the name of an assigned IPv6 prefix pool configured on the cnBNG. This field is not terminated by NULL (hexadecimal 00).

Configure Delegated-IPv6-Prefix-Pool attribute

You can configure the **Delegated-IPv6-Prefix-Pool** attribute using the user-profile on the RADIUS server. You can then directly receive it from the RADIUS server. No configuration is needed on the cnBNG CP side.

Procedure

Step 1 Define the **Delegated-IPv6-Prefix-Pool** attribute on the RADIUS server.

Example:

```
user@example.com    Password="abc"
    Service-Type=Framed-User,
    Delegated-IPv6-Prefix-Pool="dhcpv6-iapd-pool"
```

Step 2 Use the **show subscriber session detail** command to verify the configuration.

Example:

```
bng# show subscriber session detail
Mon Apr  7  10:05:13.666 UTC+00:00
subscriber-details
{
  "subResponses": [
    {
      "subLabel": "16777219",
      "mac": "cc11.0000.0001",
      "acct-sess-id": "01000003",
      "upf": "asr9k-1",
      "port-id": "Bundle-Ether1.100",
      "up-subs-id": "3",
      "sesstype": "ppp",
      "state": "established",
      "subCreateTime": "Mon, 07 Apr 2025 10:04:53 UTC",
      "subsAttr": {
        "attrs": {
          "Authentic": "RADIUS(1)",
          "Framed-Protocol": "PPP(1)",
          "addr": "33.0.0.4",
          "addrv6": "2:3::202",
          "circuit-id-tag": "circuit1",
          "client-mac-address": "cc11.0000.0001",
          "connect-progress": "IPV6CP Open",
          "delegated-prefix": "3002:db0:0:1::/64",
          "dhcpv6-client-id": "0x00030001cc1100000001",
          "inner-vlan-id": "200",
          "outer-vlan-id": "100",
        }
      },
      "subcfgInfo": {
        "committedAttrs": {
          "attrs": {
            "Stateful-IPv6-Address-Pool": "dhcpv6-iana-pool",
            "accounting-list": "aaa-prof1",
            "acct-interval": "2000",
            "addr-pool": "ipv4-pool",
            "delegated-ipv6-pool": "dhcpv6-iapd-pool",
            "session-acct-enabled": "true",
            "vrf": "default"
```