IPv6 Client IP Address Learning

Information About IPv6 Client Address Learning

Client Address Learning is configured on device to learn the IPv4 and IPv6 address of wireless client, and the client's transition state maintained by the device on association and timeout.

There are three ways for an IPv6 client to acquire IPv6 addresses:

- Stateless Address Auto-Configuration (SLAAC)
- Stateful DHCPv6
- Static Configuration

In all of these methods, the IPv6 client always sends a neighbor solicitation Duplicate Address Detection (DAD) request to ensure that there is no duplicate IP address on the network. The device snoops on the Neighbor Discovery Protocol (NDP) and DHCPv6 packets of the client to learn about its client IP addresses.

Address Assignment Using SLAAC

The most common method for IPv6 client address assignment is SLAAC, which provides simple plug-and-play connectivity, where clients self-assign an address based on the IPv6 prefix.

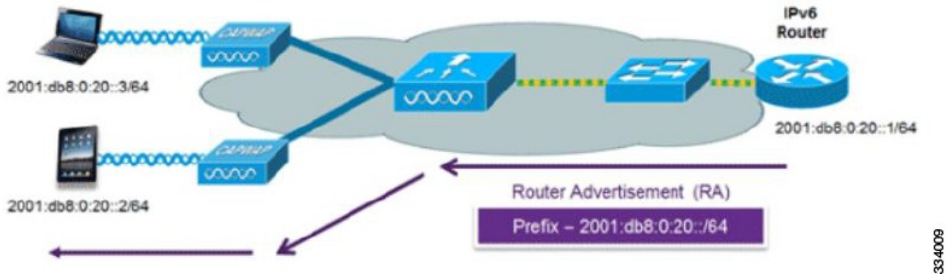
SLAAC is configured as follows:

- A host sends a Router Solicitation message.
- The host waits for a Router Advertisement message.
- The host take the first 64 bits of the IPv6 prefix from the Router Advertisement message and combines it with the 64 bit EUI-64 address (in the case of Ethernet, this is created from the MAC address) to create a global unicast message. The host also uses the source IP address, in the IP header, of the Router Advertisement message, as its default gateway.
- Duplicate Address Detection is performed by the IPv6 clients to ensure that random addresses that are picked do not collide with other clients.

Note The last 64 bits of the IPv6 address can be learned by using one of the following algorithms:

- EUI-64, which is based on the MAC address of the interface
- Private addresses that are randomly generated

Figure 1: Address Assignment Using SLAAC



The following Cisco IOS configuration commands from a Cisco-capable IPv6 router are used to enable SLAAC addressing and router advertisements:

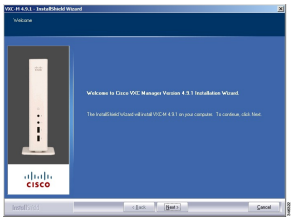
```
ipv6 unicast-routing
interface Vlan20
description IPv6-SLAAC
ip address 192.168.20.1 255.255.255.0
ipv6 address FE80:DB8:0:20::1 linklocal
ipv6 address 2001:DB8:0:20::1/64
ipv6 enable
end
```

Stateful DHCPv6 Address Assignment

The use of DHCPv6 is not required for IPv6 client connectivity if SLAAC is already deployed. There are two modes of operation for DHCPv6, that is, Stateless and Stateful.

The DHCPv6 Stateless mode is used to provide clients with additional network information that is not available in the router advertisement, but not an IPv6 address, becuase this is already provided by SLAAC. This information includes the DNS domain name, DNS servers, and other DHCP vendor-specific options.

Figure 2: Stateful DHCPv6 Address Assignment



The following interface configuration is for a Cisco IOS IPv6 router implementing stateless DHCPv6 with SLAAC enabled:

```
ipv6 unicast-routing
ipv6 dhcp pool IPV6_DHCPPOOL
address prefix 2001:db8:5:10::/64
domain-name cisco.com
dns-server 2001:db8:6:6::1
interface Vlan20
description IPv6-DHCP-Stateless
ip address 192.168.20.1 255.255.255.0
ipv6 nd other-config-flag
ipv6 dhcp server IPV6_DHCPPOOL
ipv6 address 2001:DB8:0:20::1/64
end
```

Router Solicitation

A Router Solicitation message is issued by a host controller to facilitate local routers to transmit a Router Advertisement from which the controller can obtain information about local routing, or perform stateless auto configuration. Router Advertisements are transmitted periodically and the host prompts with an immediate Router Advertisement using a Router Solicitation such as - when it boots or following a restart operation.

Router Advertisement

A Router Advertisement message is issued periodically by a router or in response to a Router Solicitation message from a host. The information contained in these messages is used by a host to perform stateless auto configuration and to modify its routing table.

Neighbor Discovery

IPv6 Neighbor Discovery is a set of messages and processes that determine relationships between neighboring nodes. Neighbor Discovery replaces the Address Resolution Protocol (ARP), Internet Control Message Protocol (ICMP) Router Discovery, and ICMP Redirect used in IPv4.

IPv6 Neighbor Discovery inspection analyzes neighbor discovery messages in order to build a trusted binding table database, and IPv6 Neighbor Discovery packets that do not comply, are dropped. The neighbor binding table in the tracks each IPv6 address and its associated MAC address. Clients are removed from the table according to neighbor-binding timers.

Neighbor Discovery Suppression

The IPv6 addresses of wireless clients are cached by a device once the wireless client is in RUN state. When the device receives an NS multicast, it looks into the IPv6 addresses cached. If the target address is known to the device and belongs to one of its wireless clients, the device converts the NS from multicast to unicast and forward it to the wireless client. If the target address is not present in the cache, then device interprets that the Multicast NS is for a wired entity and forward it towards the wired side and not to the wireless client.

The same behavior is seen for ARP request in case of IPv4 address, where the device maintains IPv4 address of the wireless client in the cache.

When neither of the configuration is enabled, and when the device receives Non-DAD or DAD NS multicast looking for an IPv6 address, and if the target address is known to the device and belongs to one of its clients, the device will convert the multicast NS to unicast NS, with the destination MAC address, replaced with client's MAC and forward the unicast packet towards client.

When full-proxy is enabled, and when the device receives Non-DAD or DAD NS multicast, looking for an IPv6 address, and if the target address is known to the device and belongs to one of its clients, the device will reply with an NA message on behalf of the client.

You can use the **ipv6 nd proxy** command to enable or disable DAD or full proxy.

When the device receives an DAD-NS multicast looking for an IPv6 address, and if the target address is known to the device and belongs to one of its clients, the device will reply with an NA message on behalf of the client.

When the device receives Non-DAD NS multicast looking for an IPv6 address, and if the target address is known to the device and belongs to one of its clients, the device will convert the multicast NS to unicast NS, with the destination MAC address, replaced with client's MAC and forward the unicast packet towards client.

If the device does not have the IPv6 address of a wireless client, the device does not respond with NA; instead, it forwards the NS packet to the wired side. Reason for forwarding to Wired Side is due to the assumption that all wireless client IPv6 address and the its mapped MAC address should be available in the controller and if an IPv6 address required in the NS is not available, then that address is not a wireless client address, so forwarded to wired side.

Router Advertisement Guard

The RA Guard feature increases the security of the IPv6 network by dropping router advertisements coming from wireless clients. Without this feature, misconfigured or malicious IPv6 clients could announce themselves as a router for the network, often with a high priority, which could take precedence over legitimate IPv6 routers. By default, RA guard is always enabled on the controller.

- Port on which the frame is received
- IPv6 source address
- Prefix list
- Trusted or Untrusted ports for receiving the router advertisement guard messages
- Trusted/Untrusted IPv6 source addresses of the router advertisement sender
- Trusted/Untrusted Prefix list and Prefix ranges
- Router preference

Router Advertisement Throttling

RA throttling allows the controller to enforce limits to the RA packets headed toward the wireless network. By enabling RA throttling, routers that send multiple RA packets can be trimmed to a minimum frequency that will still maintain an IPv6 client connectivity. If a client sends an RS packet, an RA is sent back to the client. This RA is allowed through the controller and unicast to the client. This process ensures that the new clients or roaming clients are not affected by the RA throttling.

Prerequisites for IPv6 Client Address Learning

Before configuring IPv6 client address learning, configure the clients to support IPv6.

To enable wireless IPv6 client connectivity, the underlying wired network must support IPv6 routing and an address assignment mechanism, such as SLAAC or DHCPv6. The wireless LAN controller must have L2 adjacency to the IPv6 router.

Note The AP learns IPv6 client address based on source IP address even though Neighbor Advertisements can hold rest of the IPv6 addresses. AP won't look into the Neighbor Advertisements to learn the IPv6 address learnt by the client. This behavior is seen only on Apple clients and not on Microsoft Windows clients.

IPv6 Address Tracking for Wireless Clients

Until Cisco IOS XE 17.9.1, the controller supported a maximum of eight IPv6 addresses per wireless client. After eight IPv6 addresses were learnt for a wireless client, the controller dropped that wireless client's data traffic coming with new IPv6 source addresses.

However, in Cisco IOS XE 17.9.2 release, the controller allows data traffic of the wireless clients coming with new IPv6 source addresses even after eight addresses have been learnt for respective wireless clients. The controller continues to learn new iPv6 addresses of the wireless clients from the wireless clients' control traffic (IPv6 NS/NA and DHCPv6), and keeps track of only a maximum of eight addresses per wireless client.

To allow forwarding of the multicast neighbor solicitation (NS) queries for unknown IPv6 target addresses of wireless clients (the client addresses that are not tracked by the controller) to wireless clients, the **wireless ipv6 nd ns-forward** configuration must be enabled.

Important We recommend that you configure IPv6 Multicast over Multicast (MoM) tunnel along with the **wireless ipv6 nd ns-forward** configuration.

Note In Cisco IOS XE 17.9.2, since the controller allows IPv6 traffic without address tracking beyond the eight IPv6 address limit, some of the features such as, User Defined Network, iPSK Peer-to-Peer Blocking, Management over Wireless, Neighbor Discovery Suppression, IP Theft Detection, and so on, may not work for the wireless clients using more than eight addresses. You can disable the new behavior by enabling the IP Source Guard feature https://www.cisco.com/c/en/us/td/docs/wireless/controller/9800/17-9/config-guide/b_wl_17_9_cg/m_ipsg_ewlc.html?bookSearch=true.

Configuring Unknown Address Multicast Neighbor Solicitation Forwarding

To allow forwarding of the multicast neighbor solicitation (NS) queries for unknown IPv6 target addresses of wireless clients (the client addresses that are not tracked by the controller) to wireless clients, perform the following steps:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless ipv6 nd ns-forward** **Example:** `Device (config)# wireless ipv6 nd ns-forward`	Enables forwarding of the multicast neighbor solicitation (NS) messages for unknown IPv6 target addresses of wireless clients (the client addresses that are not tracked by the controller) to wireless clients. **Note** **We recommend that you configure IPv6 Multicast over Multicast (MoM) tunnel along with this configuration.**

Configuring RA Throttle Policy (CLI)

Configure RA Throttle policy to allow the enforce the limits

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**ipv6 nd ra-throttler policy ra-throttler1** **Example:** `Device(config)# ipv6 nd ra-throttler policy ra-throttler1`	Define the router advertisement (RA) throttler policy name and enter IPv6 RA throttle policy configuration mode.
Step 3	**throttleperiod 500** **Example:** `Device(config-nd-ra-throttle)# throttle-period 500`	Configures the throttle period in an IPv6 RA throttler policy. Throttle period is in seconds and it is the time while the controller will not forward RA to the wireless clients.
Step 4	**max-through 10** **Example:** `Device(config-nd-ra-throttle)# max-through 15`	Limits multicast RAs per VLAN per throttle period.
Step 5	**allow-atleast 5 at-most 10** **Example:** `Device(config-nd-ra-throttle)# allow at-least 5 at-most 10`	Limits the number of multicast RAs per device per throttle period in an RA throttler policy.

Applying RA Throttle Policy on VLAN (GUI)

Procedure

Step 1 Choose **Configuration** > **Services** > **RA Throttle Policy**.

Step 2 Click **Add**. The **Add RA Throttle Policy** dialog box appears.

Step 3 Enter a name for the policy in the **Name** field.

Step 4 Choose the desired option from the **Medium Type** drop-down list.

Step 5 Enter a value in the **Throttle Period** field. RA throttling takes place only after the Max Through limit is reached for the VLAN or the Allow At-Most value is reached for a particular router.

Step 6 Enter a value for the **Max Through** field, which is the maximum number of RA packets on a VLAN that can be sent before throttling takes place. The **No Limit** option allows an unlimited number of RA packets through with no throttling.

Step 7 Choose an **Interval Option**, which allows the device to act differently based on the RFC 3775 value set in IPv6 RA packets, from the following options:

- Ignore—Causes the RA throttle to treat packets with the interval option as a regular RA and subject to throttling if in effect.
- Passthrough—Allows any RA messages with the RFC 3775 interval option to go through without throttling.

- Throttle—Causes the RA packets with the interval option to always be subject to rate limiting.

Step 8 Enter the minimum number of RA packets per router that can be sent as multicast before throttling takes place in the **At Least Multicast RAs** field.

Step 9 Enter the maximum number of RA packets per router that can be sent as multicast before throttling takes place in the **At Most Multicast RAs** field. The **No Limit** option allows an unlimited number of RA packets through the router.

Step 10 Click the **Add & Apply to Device** button.

Applying RA Throttle Policy on a VLAN (CLI)

Applying the RA Throttle policy on a VLAN. By enabling RA throttling, routers that send many RA packets can be trimmed to a minimum frequency that will still maintain an IPv6 client connectivity.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**vlan configuration 1** **Example:** `Device(config)# vlan configuration 1`	Configures a VLAN or a collection of VLANs and enters VLAN configuration mode.
Step 3	**ipv6 nd ra throttler attach-policy ra-throttler1** **Example:** `Device(config-vlan)# ipv6 nd ra throttler attach-policy ra-throttler1`	Attaches an IPv6 RA throttler policy to a VLAN or a collection of VLANs.

Configuring IPv6 Interface on a Switch (GUI)

Procedure

Step 1 Choose **Configuration** > **Layer2** > **VLAN** > **SVI**.

Step 2 Click **Add**.

Step 3 Enter **VLAN Number**, **Description** and **MTU (Bytes)**.

Step 4 Enable or disable the **Admin Status** toggle button.

Step 5 In **IP Options**, check the **IPv6** check box.

Step 6 Choose the type of **Static** address from the drop-down list and enter the Static Address.

Step 7 Check or uncheck the **DHCP**, **Autoconfig** and **Act as an IPv6 DHCP client** check boxes.

If you check the **DHCP** check box, the **Rapid Commit** check box is displayed. Check or uncheck the **Rapid Commit** check box.

Step 8 Click **Apply to Device**.

Configuring IPv6 on Interface (CLI)

Follow the procedure given below to configure IPv6 on an interface:

Before you begin

Enable IPv6 on the client and IPv6 support on the wired infrastructure.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface vlan** *vlan-id* **Example:** `Device(config)# interface vlan 10`	Creates an interface and enters interface configuration mode.

Step 4	**ip address fe80::1 link-local**
	Example:
	`Device(config-if)# ip address 198.51.100.1 255.255.255.0`
	`Device(config-if)# ipv6 address fe80::1 link-local`
	`Device(config-if)# ipv6 address 2001:DB8:0:1:FFFF:1234::5/64`
	`Device(config-if)# ipv6 address 2001:DB8:0:0:E000::F/64`
		Configures IPv6 address on the GigabitEthernet interface using the link-local option.

Step 5	**ipv6 enable** **Example:**	(Optional) Enables IPv6 on the GigabitEthernet interface.

	Command or Action	Purpose
	`Device(config)# ipv6 enable`	
Step 6	**end** **Example:** `Device(config)# end`	Exits interface mode.

Configuring DHCP Pool on Switch (GUI)

Procedure

Step 1 Choose **Administration** > **DHCP**.
Step 2 Click the **Add** button. The **Create DHCP Pool** dialog box appears.
Step 3 Enter a pool name in the **DHCP Pool Name** field. The name must not be greater than 236 characters in length.
Step 4 Choose either **IPv4** or **IPv6** from the **IP Type** drop-down list.
Step 5 Enter an IP address in the **Network** field.
Step 6 Choose any one of the available subnet masks from the **Subnet Mask** drop-down list.
Step 7 Enter an IP address in the **Starting ip** field.
Step 8 Enter an IP address in the **Ending ip** field.
Step 9 Optional, set the status of the **Reserved Only** field to **Enabled** if you wish to reserve the DHCP pool.
Step 10 Choose the desired option from the **Lease** drop-down list.
Step 11 Selecting the **User Defined** option from the **Lease** drop-down list enables the **(0-365 days)**, **(0-23 hours)**, and **(0-59 minutes)** fields. Enter appropriate values.
Step 12 Click the **Save & Apply to Device** button.
Step 13 For IPv6, Enter the **DNS Server, DNS Domain Name, and Ipv6 Address Allocation**.

Configuring DHCP Pool on Switch (CLI)

Follow the procedure given below to configure DHCP Pool on an interface:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 dhcp pool** *vlan-id* **Example:** `Device(config)# ipv6 dhcp pool 21`	Enters the configuration mode and configures the IPv6 DHCP pool on the Vlan.
Step 4	**address prefix 2001:DB8:0:1:FFFF:1234::/64 lifetime 300 10** **Example:** `Device(config-dhcpv6)# address prefix 2001:DB8:0:1:FFFF:1234::/64 lifetime 300 10`	Enters the configuration-dhcp mode and configures the address pool and its lifetime on a Vlan.
Step 5	**dns-server 2001:100:0:1::1** **Example:** `Device(config-dhcpv6)# dns-server 2001:20:21::1`	Configures the DNS servers for the DHCP pool.
Step 6	**domain-name example.com** **Example:** `Device(config-dhcpv6)# domain-name example.com`	Configures the domain name to complete unqualified host names.
Step 7	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Stateless Auto Address Configuration Without DHCP on Switch (CLI)

Follow the procedure given below to configure stateless auto address configuration without DHCP:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
	`Device> enable`	
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface vlan 1** **Example:** `Device(config)# interface vlan 1`	Creates an interface and enters interface configuration mode.

Step 4	**ip address fe80::1 link-local**
	Example:
	`Device(config-if)# ip address 198.51.100.1 255.255.255.0`
	`Device(config-if)# ipv6 address fe80::1 link-local`
	`Device(config-if)# ipv6 address 2001:DB8:0:1:FFFF:1234::5/64`
	`Device(config-if)# ipv6 address 2001:DB8:0:0:E000::F/64`
		Configures IPv6 address on the GigabitEthernet interface using the link-local option.

Step 5	**ipv6 enable** **Example:** `Device(config)# ipv6 enable`	(Optional) Enables IPv6 on the GigabitEthernet interface.

Step 6	**no ipv6 nd managed-config-flag**
	Example:
	`Device(config)# interface vlan 1`
	`Device(config-if)# no ipv6 nd managed-config-flag`
		Ensures the attached hosts do not use stateful autoconfiguration to obtain addresses.

Step 7	**no ipv6 nd other-config-flag** **Example:** `Device(config-if)# no ipv6 nd other-config-flag`	Ensures the attached hosts do not use stateful autoconfiguration to obtain non-address options from DHCP (domain etc).
Step 8	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Stateless Auto Address Configuration With DHCP on Switch

Follow the procedure given below to configure stateless auto address configuration with DHCP:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface vlan 1** **Example:** `Device(config)# interface vlan 1`	Creates an interface and enters interface configuration mode.

Step 4	**ip address fe80::1 link-local**
	Example:
	`Device(config-if)# ip address 198.51.100.1 255.255.255.0`
	`Device(config-if)# ipv6 address fe80::1 link-local`
	`Device(config-if)# ipv6 address 2001:DB8:0:1:FFFF:1234::5/64`
	`Device(config-if)# ipv6 address 2001:DB8:0:0:E000::F/64`
		Configures IPv6 address on the GigabitEthernet interface using the link-local option.

Step 5	**ipv6 enable** **Example:** `Device(config)# ipv6 enable`	(Optional) Enables IPv6 on the GigabitEthernet interface.
Step 6	ipv6 nd prefix ipaddress **Example:** `ipv6 nd prefix 2001:9:3:54::/64 no-advertise`	Specifies a subnet prefix.
Step 7	**no ipv6 nd managed-config-flag** **Example:**	Ensures the attached hosts do not use stateful autoconfiguration to obtain addresses.

	Command or Action	Purpose
	`Device(config)# interface vlan 1` `Device(config-if)# no ipv6 nd managed-config-flag`	
Step 8	**ipv6 nd other-config-flag** **Example:** `Device(config-if)# no ipv6 nd other-config-flag`	Ensures the attached hosts do not use stateful autoconfiguration to obtain non-address options from DHCP (domain etc).
Step 9	ipv6 dhcp server servername **Example:** `ipv6 dhcp server VLAN54`	Displays the configuration parameters.
Step 10	**end** **Example:** `Device(config)# end`	Exits interface mode.

Configuring Stateless Address Auto Configuration Without DHCP on Switch (CLI)

Follow the procedure given below to configure stateless auto address configuration without DHCP:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface vlan 1** **Example:** `Device(config)# interface vlan 1`	Creates an interface and enters interface configuration mode.
Step 4	**ip address fe80::1 link-local** **Example:** `Device(config-if)# ip address 198.51.100.1 255.255.255.0`	Configures IPv6 address on the GigabitEthernet interface using the link-local option.

	Command or Action	Purpose
	Device(config-if)# **ipv6 address fe80::1 link-local** Device(config-if)# **ipv6 address 2001:DB8:0:1:FFFF:1234::5/64** Device(config-if)# **ipv6 address 2001:DB8:0:0:E000::F/64**	
Step 5	**ipv6 enable** **Example:** Device(config)# **ipv6 enable**	(Optional) Enables IPv6 on the GigabitEthernet interface.
Step 6	**no ipv6 nd managed-config-flag** **Example:** Device(config)# **interface vlan 1** Device(config-if)# **no ipv6 nd managed-config-flag**	Ensures the attached hosts do not use stateful autoconfiguration to obtain addresses.
Step 7	**no ipv6 nd other-config-flag** **Example:** Device(config-if)# **no ipv6 nd other-config-flag**	Ensures the attached hosts do not use stateful autoconfiguration to obtain non-address options from DHCP (domain etc).
Step 8	**end** **Example:** Device(config)# **end**	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Native IPv6

Information About IPv6

IPv6 is a packet-based protocol used to exchange data, voice, and video traffic over digital networks. IPv6 is based on IP, but with a much larger address space, and improvements such as a simplified main header and extension headers. The architecture of IPv6 has been designed to allow existing IPv4 users to transition easily to IPv6 while continuing to use services such as end-to-end security, quality of service (QoS), and globally unique addresses. The larger IPv6 address space allows networks to scale and provide global reachability.

Note The features and functions that work on IPv4 networks with IPv4 addresses also work on IPv6 networks with IPv6 addresses.

General Guidelines

- For IPv6 functionality to work, ensure that you disable IPv6 multicast routing.
- The Wireless Management interface should have only one static IPv6 address.

- Router advertisement should be suppressed on the wireless management interface and client VLANs (if IPv6 is configured on the client VLAN).
- Preferred mode is part of an AP join profile. When you configure the preferred mode as IPv6, an AP attempts to join over IPv6 first. If it fails, the AP falls back to IPv4.
- You should use MAC addresses for RA tracing of APs and clients.
- APs can join IPv6 controllers only with an IPv6 static address. If you have a controller with auto configurations and multiple IPv6 addresses, APs cannot join the IPv6 controllers.

Unsupported Features

- UDP Lite is not supported.
- AP sniffer over IPv6 is not supported.
- IPv6 is not supported for the HA port interface.
- Auto RF grouping over IPv6 is not supported. Only static RF grouping is supported.

Configuring IPv6 Addressing

Follow the procedure given below to configure IPv6 addressing:

Note All the features and functions that work on IPv4 networks with IPv4 addresses will work on IPv6 networks with IPv6 addresses too.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ipv6 unicast-routing** **Example:** `Device(config)# ipv6 unicast-routing`	Configures IPv6 for unicasting.
Step 3	**interface vlan 1** **Example:** `Device(config)# interface vlan 1`	Creates an interface and enters interface configuration mode.
Step 4	**ipv6 address** *ipv6-address* **Example:** `Device(config-if)# ipv6 address FD09:9:2:49::53/64`	Specifies a global IPv6 address.

	Command or Action	Purpose
Step 5	**ipv6 enable** **Example:** `Device(config-if)# ipv6 enable`	Enables IPv6 on the interface.
Step 6	**ipv6 nd ra suppress all** **Example:** `Device(config-if)# ipv6 nd ra suppress all`	Suppresses IPv6 router advertisement transmissions on the interface.
Step 7	**exit** **Example:** `Device(config-if)# exit`	Returns to global configuration mode.
Step 8	**wireless management interface gigabitEthernet** *gigabitEthernet-interface-vlan 64* **Example:** `Device(config)# wireless management interface gigabitEthernet vlan 64`	Configures the ports that are connected to the supported APs with the wireless management interface.
Step 9	**ipv6 route** *ipv6-address* **Example:** `Device(config)# ipv6 route ::/0 FD09:9:2:49::1`	Specifies IPv6 static routes.

Creating an AP Join Profile (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **AP Join**.

Step 2 On the **AP Join Profile** window, click the **General** tab and click **Add**.

Step 3 In the **Name** field enter, a name for the AP join profile.

Step 4 (Optional) Enter a description for the AP join profile.

Step 5 Choose **CAPWAP** > **Advanced**.

Step 6 Under the **Advanced** tab, from the **Preferred Mode** drop-down list, choose **IPv6**. This sets the preferred mode of APs as IPv6.

Step 7 Click **Save & Apply to Device**.

Creating an AP Join Profile (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap profile** *ap-profile* **Example:** `Device(config)# ap profile xyz-ap-profile`	Configures an AP profile and enters AP profile configuration mode.
Step 3	**description** *ap-profile-name* **Example:** `Device(config-ap-profile)# description "xyz ap profile"`	Adds a description for the AP profile.
Step 4	**preferred-mode ipv6** **Example:** `Device(config-ap-profile)# preferred-mode ipv6`	Sets the preferred mode of APs as IPv6.

Configuring the Primary and Backup Controller (GUI)

Before you begin

Ensure that you have configured an AP join profile prior to configuring the primary and backup controller s.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **AP Join**.

Step 2 On the **AP Join Profile** window, click the AP join profile name.

Step 3 In the **Edit AP Join Profile** window, click the **CAPWAP** tab.

Step 4 In the **High Availability** tab, under **Backup Controller Configuration**, check the **Enable Fallback** check box.

Step 5 Enter the primary and secondary controller names and IP addresses.

Step 6 Click **Update & Apply to Device**.

Configuring Primary and Backup Controller (CLI)

Follow the procedure given below to configure the primary and secondary controllers for a selected AP:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap profile** *profile-name* **Example:** `Device(config)# ap profile yy-ap-profile`	Configures an AP profile and enters AP profile configuration mode.
Step 3	**capwap backup primary** *primary-controller-name primary-controller-ip* **Example:** `Device(config)# capwap backup primary WLAN-Controller-A 2001:DB8:1::1`	Configures AP CAPWAP parameters with the primary backup controller's name. **Note** You need to enable fast heartbeat for **capwap backup primary** and **capwap backup secondary** to work. AP disconnection may occur if the link between the controller and AP is not reliable and fast heartbeat is enabled.
Step 4	**ap capwap backup secondary** *secondary-controller-name secondary-controller-ip* **Example:** `Device(config)# capwap backup secondary WLAN-Controller-B 2001:DB8:1::1`	Configures AP CAPWAP parameters with the secondary backup controller's name.
Step 5	**syslog host** *ipaddress* **Example:** `Device(config)# syslog host 2001:DB8:1::1`	Configures the system logging settings for the APs.
Step 6	**tftp-downgrade** *tftp-server-ip imagename* **Example:** `Device(config)# tftp-downgrade 2001:DB8:1::1 testimage`	Initiates AP image downgrade from a TFTP server for all the APs.

Verifying IPv6 Configuration

Use the following **show** command to verify the IPv6 configuration:

```
Device# show wireless interface summary

Wireless Interface Summary
```

```
Interface Name Interface Type VLAN ID IP Address      IP Netmask     MAC Address
-------------------------------------------------------------------------------------
Vlan49         Management     49      0.0.0.0       255.255.255.0    001e.f64c.1eff
                                      fd09:9:2:49::54/64
```