Configuring IPv6 First Hop Security

Prerequisites for First Hop Security in IPv6

- You have configured the necessary IPv6 enabled SDM template.
- You should be familiar with the IPv6 neighbor discovery feature.

Restrictions for First Hop Security in IPv6

- The following restrictions apply when applying FHS policies to EtherChannel interfaces (Port Channels):
  - A physical port with an FHS policy attached cannot join an EtherChannel group.
  - An FHS policy cannot be attached to an physical port when it is a member of an EtherChannel group.
- By default, a snooping policy has a security-level of guard. When such a snooping policy is configured on an access switch, external IPv6 Router Advertisement (RA) or Dynamic Host Configuration Protocol

server packets are blocked, even though the uplink port facing the router or DHCP server/relay is configured as a trusted port. To allow IPv6 RA or DHCPv6 server messages, do the following:

- Apply an IPv6 RA-guard policy (for RA) or IPv6 DHCP-guard policy (for DHCP server messages) on the uplink port.
- Configure a snooping policy with a lower security-level, for example glean or inspect. However; configuring a lower security level is not recommended with such a snooping policy, because benefits of First Hop security features are not effective.

Information about First Hop Security in IPv6

First Hop Security in IPv6 (FHS IPv6) is a set of IPv6 security features, the policies of which can be attached to a physical interface, an EtherChannel interface, or a VLAN. An IPv6 software policy database service stores and accesses these policies. When a policy is configured or modified, the attributes of the policy are stored or updated in the software policy database, then applied as was specified. The following IPv6 policies are currently supported:

- IPv6 Snooping Policy—IPv6 Snooping Policy acts as a container policy that enables most of the features available with FHS in IPv6.
- IPv6 FHS Binding Table Content—A database table of IPv6 neighbors connected to the switch is created from information sources such as Neighbor Discovery (ND) protocol snooping. This database, or binding, table is used by various IPv6 guard features (such as IPv6 ND Inspection) to validate the link-layer address (LLA), the IPv4 or IPv6 address, and prefix binding of the neighbors to prevent spoofing and redirect attacks.
- IPv6 Neighbor Discovery Inspection—IPv6 ND inspection learns and secures bindings for stateless autoconfiguration addresses in Layer 2 neighbor tables. IPv6 ND inspection analyzes neighbor discovery messages in order to build a trusted binding table database and IPv6 neighbor discovery messages that do not conform are dropped. An ND message is considered trustworthy if its IPv6-to-Media Access Control (MAC) mapping is verifiable.

  This feature mitigates some of the inherent vulnerabilities of the ND mechanism, such as attacks on DAD, address resolution, router discovery, and the neighbor cache.

  

  Note Effective Cisco IOS XE Release 16.3.1, ND Inspection functionality, IPv6 Snooping Policy, and IPv6 FHS Binding Table Content are supported through Switch Integrated Security Feature (SISF)-based Device Tracking. For more information, see Configuring SISF based device tracking section of the Software Configuration Guide.

- IPv6 Router Advertisement Guard—The IPv6 Router Advertisement (RA) guard feature enables the network administrator to block or reject unwanted or rogue RA guard messages that arrive at the network switch platform. RAs are used by routers to announce themselves on the link. The RA Guard feature analyzes the RAs and filters out bogus RAs sent by unauthorized routers. In host mode, all router advertisement and router redirect messages are disallowed on the port. The RA guard feature compares configuration information on the Layer 2 device with the information found in the received RA frame. Once the Layer 2 device has validated the content of the RA frame and router redirect frame against the

configuration, it forwards the RA to its unicast or multicast destination. If the RA frame content is not validated, the RA is dropped.

- IPv6 DHCP Guard—The IPv6 DHCP Guard feature blocks reply and advertisement messages that come from unauthorized DHCPv6 servers and relay agents. IPv6 DHCP guard can prevent forged messages from being entered in the binding table and block DHCPv6 server messages when they are received on ports that are not explicitly configured as facing a DHCPv6 server or DHCP relay. To use this feature, configure a policy and attach it to an interface or a VLAN. To debug DHCP guard packets, use the debug ipv6 snooping dhcp-guard privileged EXEC command.
- IPv6 Source Guard—Like IPv4 Source Guard, IPv6 Source Guard validates the source address or prefix to prevent source address spoofing.

  A source guard programs the hardware to allow or deny traffic based on source or destination addresses. It deals exclusively with data packet traffic.

  To debug source-guard packets, use the debug ipv6 snooping source-guard privileged EXEC command.

  The following restrictions apply:

  - An FHS policy cannot be attached to an physical port when it is a member of an EtherChannel group.
  - When IPv6 source guard is enabled on a switch port, NDP or DHCP snooping must be enabled on the interface to which the switch port belongs. Otherwise, all data traffic from this port will be blocked.
  - An IPv6 source guard policy cannot be attached to a VLAN. It is supported only at the interface level.
  - When you configure IPv4 and IPv6 source guard together on an interface, it is recommended to use ip verify source mac-check instead of ip verify source . IPv4 connectivity on a given port might break due to two different filtering rules set — one for IPv4 (IP-filter) and the other for IPv6 (IP-MAC filter).
  - You cannot use IPv6 Source Guard and Prefix Guard together. When you attach the policy to an interface, it should be "validate address" or "validate prefix" but not both.
  - PVLAN and Source/Prefix Guard cannot be applied together.

  For more information on IPv6 Source Guard, see the IPv6 Source Guard chapter of the Cisco IOS IPv6 Configuration Guide Library on Cisco.com.
- IPv6 Prefix Guard—The IPv6 prefix guard feature works within the IPv6 source guard feature, to enable the device to deny traffic originated from non-topologically correct addresses. IPv6 prefix guard is often used when IPv6 prefixes are delegated to devices (for example, home gateways) using DHCP prefix delegation. The feature discovers ranges of addresses assigned to the link and blocks any traffic sourced with an address outside this range.

  For more information on IPv6 Prefix Guard, see the IPv6 Prefix Guard chapter of the Cisco IOS IPv6 Configuration Guide Library on Cisco.com.
- IPv6 Destination Guard—The IPv6 destination guard feature works with IPv6 neighbor discovery to ensure that the device performs address resolution only for those addresses that are known to be active on the link. It relies on the address glean functionality to populate all destinations active on the link into the binding table and then blocks resolutions before they happen when the destination is not found in the binding table.

For more information about IPv6 Destination Guard, see the IPv6 Destination Guard chapter of the Cisco IOS IPv6 Configuration Guide Library on Cisco.com.

How to Configure an IPv6 Snooping Policy

The IPv6 Snooping Policy feature has been deprecated. Although the commands are visible on the CLI and you can configure them, we recommend that you use the Switch Integrated Security Feature (SISF)-based Device Tracking feature instead.

Beginning in privileged EXEC mode, follow these steps to configure IPv6 Snooping Policy :

Procedure

	Command or Action	Purpose
Step 1	configure terminal **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 2	ipv6 snooping policy policy-name **Example:** `Device(config)# ipv6 snooping policy example_policy`	Creates a snooping policy and enters IPv6 Snooping Policy Configuration mode.
Step 3	{[default] | [device-role {node | switch}] | [limit address-count value] | [no] | [protocol {dhcp | ndp}] | [security-level {glean | guard | inspect}] | [tracking {disable [stale-lifetime [seconds | infinite] | enable [reachable-lifetime [seconds | infinite] }] | [trusted-port] } **Example:** Device **(config-ipv6-snooping)# security-level inspect** **Example:** Device **(config-ipv6-snooping)# trusted-port**	Enables data address gleaning, validates messages against various criteria, specifies the security level for messages. • (Optional) default—Sets all to default options. • (Optional) device-role{node] | switch}—Specifies the role of the device attached to the port. Default is node. • (Optional) limit address-count value—Limits the number of addresses allowed per target. • (Optional) no—Negates a command or sets it to defaults. • (Optional) protocol{dhcp | ndp}—Specifies which protocol should be redirected to the snooping feature for analysis. The default, is dhcp and ndp. To change the default, use the no protocol command. • (Optional) security-level{glean|guard|inspect}—Specifies

	Command or Action	Purpose
		the level of security enforced by the feature. Default is guard. glean—Gleans addresses from messages and populates the binding table without any verification. guard—Gleans addresses and inspects messages. In addition, it rejects RA and DHCP server messages. This is the default option. inspect—Gleans addresses, validates messages for consistency and conformance, and enforces address ownership. • (Optional) tracking {disable | enable}—Overrides the default tracking behavior and specifies a tracking option. • (Optional) trusted-port—Sets up a trusted port. It disables the guard on applicable targets. Bindings learned through a trusted port have preference over bindings learned through any other port. A trusted port is given preference in case of a collision while making an entry in the table.
Step 4	end **Example:** `Device(config-ipv6-snooping)# exit`	Exits configuration modes to Privileged EXEC mode.
Step 5	show ipv6 snooping policy policy-name **Example:** `Device#show ipv6 snooping policy example_policy`	Displays the snooping policy configuration.

What to do next

Attach an IPv6 Snooping policy to interfaces or VLANs.

How to Attach an IPv6 Snooping Policy to an Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Snooping policy on an interface or VLAN:

Procedure

	Command or Action	Purpose
Step 1	configure terminal **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 2	interface Interface_type stack/module/port **Example:** `Device(config)# interface gigabitethernet 1/1/4`	Specifies an interface type and identifier; enters the interface configuration mode.
Step 3	switchport **Example:** `Device(config-if)# switchport`	Enters the Switchport mode. **Note** To configure Layer 2 parameters, if the interface is in Layer 3 mode, you must enter the switchport interface configuration command without any parameters to put the interface into Layer 2 mode. This shuts down the interface and then re-enables it, which might generate messages on the device to which the interface is connected. When you put an interface that is in Layer 3 mode into Layer 2 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration. The command prompt displays as (config-if)# in Switchport configuration mode.

Step 4	ipv6 snooping [attach-policy policy_name [vlan {vlan_id | add vlan_ids | exceptvlan_ids | none | remove vlan_ids}] | vlan {vlan_id | add vlan_ids | exceptvlan_ids | none | remove vlan_ids | all}]
	Example:
	`Device(config-if)# ipv6 snooping`
	`or`
	`Device(config-if)# ipv6 snooping attach-policy example_policy`
	`or`
	`Device(config-if)# ipv6 snooping vlan 111,112`
	`or`
		Attaches a custom ipv6 snooping policy to the interface or the specified VLANs on the interface. To attach the default policy to the interface, use the ipv6 snooping command without the attach-policy keyword. To attach the default policy to VLANs on the interface, use the ipv6 snooping vlan command. The default policy is, security-level guard, device-role node, protocol ndp and dhcp.

	Command or Action	Purpose
	`Device(config-if)# ipv6 snooping attach-policy example_policy vlan 111,112`	
Step 5	do show running-config **Example:** `Device#(config-if)# do show running-config`	Verifies that the policy is attached to the specified interface without exiting the interface configuration mode.

How to Attach an IPv6 Snooping Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Snooping policy on an EtherChannel interface or VLAN:

Procedure

	Command or Action	Purpose
Step 1	configure terminal **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 2	interface range Interface_name **Example:** `Device(config)# interface range Po11`	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. **Tip** Enter the do show interfaces summary command for quick reference to interface names and types.

Step 3	ipv6 snooping [attach-policy policy_name [vlan {vlan_ids | add vlan_ids | except vlan_ids | none | remove vlan_ids | all}] | vlan [{vlan_ids | add vlan_ids | exceptvlan_ids | none | remove vlan_ids | all}]
	Example:
	`Device(config-if-range)# ipv6 snooping attach-policy example_policy`
	`or`
	`Device(config-if-range)# ipv6 snooping attach-policy example_policy vlan 222,223,224`
	or
		Attaches the IPv6 Snooping policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.

	Command or Action	Purpose
	`Device(config-if-range)#ipv6 snooping vlan 222, 223,224`	
Step 4	do show running-config interfaceportchannel_interface_name **Example:** `Device#(config-if-range)# do show running-config int po11`	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Snooping Policy to VLANs Globally

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Snooping Policy to VLANs across multiple interfaces:

Procedure

	Command or Action	Purpose
Step 1	configure terminal **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 2	vlan configuration vlan_list **Example:** `Device(config)# vlan configuration 333`	Specifies the VLANs to which the IPv6 Snooping policy will be attached ; enters the VLAN interface configuration mode.
Step 3	ipv6 snooping [attach-policy policy_name] **Example:** `Device(config-vlan-config)#ipv6 snooping attach-policy example_policy`	Attaches the IPv6 Snooping policy to the specified VLANs across all switch and stack interfaces. The default policy is attached if the attach-policy option is not used. The default policy is, security-level guard, device-role node, protocol ndp and dhcp.
Step 4	do show running-config **Example:** `Device#(config-if)# do show running-config`	Verifies that the policy is attached to the specified VLANs without exiting the interface configuration mode.

How to Configure the IPv6 Binding Table Content

Beginning in privileged EXEC mode, follow these steps to configure IPv6 Binding Table Content :

Procedure

	Command or Action	Purpose
Step 1	configure terminal **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 2	[no] ipv6 neighbor binding [vlan vlan-id {ipv6-address interface interface_type stack/module/port hw_address [reachable-lifetimevalue [seconds | default | infinite] | [tracking{ [default | disable] [reachable-lifetimevalue [seconds | default | infinite] | [enable [reachable-lifetimevalue [seconds | default | infinite] | [retry-interval {seconds| default [reachable-lifetimevalue [seconds | default | infinite] }] **Example:** `Device(config)# ipv6 neighbor binding`	Adds a static entry to the binding table database. **Note** Switch adds small variance to configured reachable-time value to improve system stability during timer expiry of binding entries.
Step 3	[no] ipv6 neighbor binding max-entries number [mac-limit number | port-limit number [mac-limit number] | vlan-limit number [[mac-limit number] | [port-limit number [mac-limitnumber]]]] **Example:** `Device(config)# ipv6 neighbor binding max-entries 30000`	Specifies the maximum number of entries that are allowed to be inserted in the binding table cache.
Step 4	ipv6 neighbor binding logging **Example:** `Device(config)# ipv6 neighbor binding logging`	Enables the logging of binding table main events.
Step 5	exit **Example:** `Device(config)# exit`	Exits global configuration mode, and places the router in privileged EXEC mode.
Step 6	show ipv6 neighbor binding **Example:** `Device# show ipv6 neighbor binding`	Displays contents of a binding table.

How to Configure an IPv6 Neighbor Discovery Inspection Policy

Beginning in privileged EXEC mode, follow these steps to configure an IPv6 ND Inspection Policy:

Procedure

	Command or Action	Purpose
Step 1	configure terminal **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 2	[no]ipv6 nd inspection policy policy-name **Example:** `Device(config)# ipv6 nd inspection policy example_policy`	Specifies the ND inspection policy name and enters ND Inspection Policy configuration mode.
Step 3	device-role {host | switch} **Example:** `Device(config-nd-inspection)# device-role switch`	Specifies the role of the device attached to the port. The default is host.
Step 4	limit address-count value **Example:** `Device(config-nd-inspection)# limit address-count 1000`	Enter 1–10,000.
Step 5	tracking {enable [reachable-lifetime {value | infinite}] | disable [stale-lifetime {value | infinite}]} **Example:** `Device(config-nd-inspection)# tracking disable stale-lifetime infinite`	Overrides the default tracking policy on a port.
Step 6	trusted-port **Example:** `Device(config-nd-inspection)# trusted-port`	Configures a port to become a trusted port.
Step 7	validate source-mac **Example:** `Device(config-nd-inspection)# validate source-mac`	Checks the source media access control (MAC) address against the link-layer address.
Step 8	no {device-role | limit address-count | tracking | trusted-port | validate source-mac} **Example:** `Device(config-nd-inspection)# no validate source-mac`	Remove the current configuration of a parameter with the no form of the command.
Step 9	default {device-role | limit address-count | tracking | trusted-port | validate source-mac} **Example:**	Restores configuration to the default values.

	Command or Action	Purpose
	`Device(config-nd-inspection)# default limit address-count`	
Step 10	do show ipv6 nd inspection policy policy_name **Example:** `Device(config-nd-inspection)# do show ipv6 nd inspection policy example_policy`	Verifies the ND Inspection Configuration without exiting ND inspection configuration mode.

How to Attach an IPv6 Neighbor Discovery Inspection Policy to an Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 ND Inspection policy to an interface or VLANs on an interface :

Procedure

	Command or Action	Purpose
Step 1	configure terminal **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 2	interface Interface_type stack/module/port **Example:** `Device(config)# interface gigabitethernet 1/1/4`	Specifies an interface type and identifier; enters the interface configuration mode.

Step 3	ipv6 nd inspection [attach-policy policy_name [vlan {vlan_ids | add vlan_ids | except vlan_ids | none | remove vlan_ids | all}] | vlan [{vlan_ids | add vlan_ids | exceptvlan_ids | none | remove vlan_ids | all}]
	Example:
	`Device(config-if)# ipv6 nd inspection attach-policy example_policy`
	`or`
	`Device(config-if)# ipv6 nd inspection attach-policy example_policy vlan 222,223,224`
	or
	`Device(config-if)# ipv6 nd inspection vlan 222, 223,224`
		Attaches the Neighbor Discovery Inspection policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.

	Command or Action	Purpose
Step 4	do show running-config **Example:** `Device#(config-if)#  do show running-config`	Verifies that the policy is attached to the specified interface without exiting the interface configuration mode.

How to Attach an IPv6 Neighbor Discovery Inspection Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Neighbor Discovery Inspection policy on an EtherChannel interface or VLAN:

Procedure

	Command or Action	Purpose
Step 1	configure terminal **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 2	interface range Interface_name **Example:** `Device(config)#  interface Po11`	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. **Tip** Enter the do show interfaces summary command for quick reference to interface names and types.

Step 3	ipv6 nd inspection [attach-policy policy_name [vlan {vlan_ids | add vlan_ids | except vlan_ids | none | remove vlan_ids | all}] | vlan [{vlan_ids | add vlan_ids | exceptvlan_ids | none | remove vlan_ids | all}]
	Example:
	`Device(config-if-range)# ipv6 nd inspection attach-policy example_policy`
	`or`
	`Device(config-if-range)# ipv6 nd inspection attach-policy example_policy vlan 222,223,224`
	or
	`Device(config-if-range)#ipv6 nd inspection vlan 222, 223,224`
		Attaches the ND Inspection policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.

	Command or Action	Purpose
Step 4	do show running-config interfaceportchannel_interface_name **Example:** `Device#(config-if-range)# do show running-config int po11`	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Neighbor Discovery Inspection Policy to VLANs Globally

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 ND Inspection policy to VLANs across multiple interfaces:

Procedure

	Command or Action	Purpose
Step 1	configure terminal **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 2	vlan configuration vlan_list **Example:** `Device(config)# vlan configuration 334`	Specifies the VLANs to which the IPv6 Snooping policy will be attached ; enters the VLAN interface configuration mode.
Step 3	ipv6 nd inspection [attach-policy policy_name] **Example:** `Device(config-vlan-config)#ipv6 nd inspection attach-policy example_policy`	Attaches the IPv6 Neighbor Discovery policy to the specified VLANs across all switch and stack interfaces. The default policy is attached if the attach-policy option is not used. The default policy is, device-role host, no drop-unsecure, limit address-count disabled, sec-level minimum is disabled, tracking is disabled, no trusted-port, no validate source-mac.
Step 4	do show running-config **Example:** `Device#(config-if)# do show running-config`	Confirms that the policy is attached to the specified VLANs without exiting the configuration mode.

How to Configure an IPv6 Router Advertisement Guard Policy

Beginning in privileged EXEC mode, follow these steps to configure an IPv6 Router Advertisement policy :

Procedure

	Command or Action	Purpose
Step 1	configure terminal **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 2	[no]ipv6 nd raguard policy policy-name **Example:** `Device(config)# ipv6 nd raguard policy example_policy`	Specifies the RA Guard policy name and enters RA Guard Policy configuration mode.
Step 3	[no]device-role {host | monitor | router | switch} **Example:** `Device(config-nd-raguard)# device-role switch`	Specifies the role of the device attached to the port. The default is host. **Note** For a network with both host-facing ports and router-facing ports, along with a RA guard policy configured with device-role host on host-facing ports or vlan, it is mandatory to configure a RA guard policy with device-role router on router-facing ports to allow the RA Guard feature to work properly.
Step 4	[no]hop-limit {maximum | minimum} value **Example:** `Device(config-nd-raguard)# hop-limit maximum 33`	(1–255) Range for Maximum and Minimum Hop Limit values. Enables filtering of Router Advertisement messages by the Hop Limit value. A rogue RA message may have a low Hop Limit value (equivalent to the IPv4 Time to Live) that when accepted by the host, prevents the host from generating traffic to destinations beyond the rogue RA message generator. An RA message with an unspecified Hop Limit value is blocked. If not configured, this filter is disabled. Configure minimum to block RA messages with Hop Limit values lower than the value you specify. Configure maximumto block RA messages with Hop Limit values greater than the value you specify.
Step 5	[no]managed-config-flag {off | on} **Example:** `Device(config-nd-raguard)# managed-config-flag on`	Enables filtering of Router Advertisement messages by the Managed Address Configuration, or "M" flag field. A rouge RA message with an M field of 1 can cause a host to use a rogue DHCPv6 server. If not configured, this filter is disabled.

	Command or Action	Purpose
		On—Accepts and forwards RA messages with an M value of 1, blocks those with 0. Off—Accepts and forwards RA messages with an M value of 0, blocks those with 1.
Step 6	[no]match {ipv6 access-list list | ra prefix-list list} **Example:** `Device(config-nd-raguard)# match ipv6 access-list example_list`	Matches a specified prefix list or access list.
Step 7	[no]other-config-flag {on | off} **Example:** `Device(config-nd-raguard)# other-config-flag on`	Enables filtering of Router Advertisement messages by the Other Configuration, or "O" flag field. A rouge RA message with an O field of 1 can cause a host to use a rogue DHCPv6 server. If not configured, this filter is disabled. On—Accepts and forwards RA messages with an O value of 1, blocks those with 0. Off—Accepts and forwards RA messages with an O value of 0, blocks those with 1.
Step 8	[no]router-preference maximum {high | medium | low} **Example:** `Device(config-nd-raguard)# router-preference maximum high`	Enables filtering of Router Advertisement messages by the Router Preference flag. If not configured, this filter is disabled. • high—Accepts RA messages with the Router Preference set to high, medium, or low. • medium—Blocks RA messages with the Router Preference set to high. • low—Blocks RA messages with the Router Preference set to medium and high.
Step 9	[no]trusted-port **Example:** `Device(config-nd-raguard)# trusted-port`	When configured as a trusted port, all attached devices are trusted, and no further message verification is performed.
Step 10	default {device-role | hop-limit {maximum | minimum} | managed-config-flag | match {ipv6 access-list | ra prefix-list } | other-config-flag | router-preference maximum| trusted-port} **Example:** `Device(config-nd-raguard)# default hop-limit`	Restores a command to its default value.

	Command or Action	Purpose
Step 11	do show ipv6 nd raguard policy policy_name **Example:** `Device(config-nd-raguard)# do show ipv6 nd raguard policy example_policy`	(Optional)—Displays the ND Guard Policy configuration without exiting the RA Guard policy configuration mode.

How to Attach an IPv6 Router Advertisement Guard Policy to an Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Router Advertisement policy to an interface or to VLANs on the interface :

Procedure

	Command or Action	Purpose
Step 1	configure terminal **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 2	interface Interface_type stack/module/port **Example:** `Device(config)# interface gigabitethernet 1/1/4`	Specifies an interface type and identifier; enters the interface configuration mode.

Step 3	ipv6 nd raguard [attach-policy policy_name [vlan {vlan_ids | add vlan_ids | except vlan_ids | none | remove vlan_ids | all}] | vlan [{vlan_ids | add vlan_ids | exceptvlan_ids | none | remove vlan_ids | all}]
	Example:
	`Device(config-if)# ipv6 nd raguard attach-policy example_policy`
	`or`
	`Device(config-if)# ipv6 nd raguard attach-policy example_policy vlan 222,223,224`
	or
	`Device(config-if)# ipv6 nd raguard vlan 222, 223,224`
		Attaches the Neighbor Discovery Inspection policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.

Step 4	do show running-config **Example:** `Device#(config-if)# do show running-config`	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Router Advertisement Guard Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Router Advertisement Guard Policy on an EtherChannel interface or VLAN:

Procedure

	Command or Action	Purpose
Step 1	configure terminal **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 2	interface range Interface_name **Example:** `Device(config)# interface Po11`	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. **Tip** Enter the do show interfaces summary command for quick reference to interface names and types.

Step 3	ipv6 nd raguard [attach-policy policy_name [vlan {vlan_ids | add vlan_ids | except vlan_ids | none | remove vlan_ids | all}] | vlan [{vlan_ids | add vlan_ids | exceptvlan_ids | none | remove vlan_ids | all}]
	Example:
	`Device(config-if-range)# ipv6 nd raguard attach-policy example_policy`
	`or`
	`Device(config-if-range)# ipv6 nd raguard attach-policy example_policy vlan 222,223,224`
	`or`
	`Device(config-if-range)#ipv6 nd raguard vlan 222, 223,224`
		Attaches the RA Guard policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.

Step 4	do show running-config interfaceportchannel_interface_name **Example:** `Device#(config-if-range)# do show running-config int po11`	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Router Advertisement Guard Policy to VLANs Globally

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Router Advertisement policy to VLANs regardless of interface:

Procedure

	Command or Action	Purpose
Step 1	configure terminal **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	vlan configuration vlan_list **Example:** `Device(config)# vlan configuration 335`	Specifies the VLANs to which the IPv6 RA Guard policy will be attached ; enters the VLAN interface configuration mode.
Step 3	ipv6 dhcp guard [attach-policy policy_name] **Example:** `Device(config-vlan-config)#ipv6 nd raguard attach-policy example_policy`	Attaches the IPv6 RA Guard policy to the specified VLANs across all switch and stack interfaces. The default policy is attached if the attach-policy option is not used.
Step 4	do show running-config **Example:** `Device#(config-if)# do show running-config`	Confirms that the policy is attached to the specified VLANs without exiting the configuration mode.

How to Configure an IPv6 DHCP Guard Policy

Beginning in privileged EXEC mode, follow these steps to configure an IPv6 DHCP (DHCPv6) Guard policy:

Procedure

	Command or Action	Purpose
Step 1	configure terminal **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 2	[no]ipv6 dhcp guard policy policy-name **Example:** `Device(config)# ipv6 dhcp guard policy example_policy`	Specifies the DHCPv6 Guard policy name and enters DHCPv6 Guard Policy configuration mode.

	Command or Action	Purpose
Step 3	[no]device-role {client | server} **Example:** `Device(config-dhcp-guard)# device-role server`	(Optional) Filters out DHCPv6 replies and DHCPv6 advertisements on the port that are not from a device of the specified role. Default is client. • client—Default value, specifies that the attached device is a client. Server messages are dropped on this port. • server—Specifies that the attached device is a DHCPv6 server. Server messages are allowed on this port.

Step 4	[no] match server access-list ipv6-access-list-name
	Example:
	`;;Assume a preconfigured IPv6 Access List as follows:`
	`Device(config)# ipv6 access-list my_acls`
	`Device(config-ipv6-acl)# permit host FE80::A8BB:CCFF:FE01:F700 any`
	`;;configure DCHPv6 Guard to match approved access list.`
	`Device(config-dhcp-guard)# match server access-list my_acls`
		(Optional). Enables verification that the advertised DHCPv6 server or relay address is from an authorized server access list (The destination address in the access list is 'any'). If not configured, this check will be bypassed. An empty access list is treated as a permit all.

Step 5	[no] match reply prefix-list ipv6-prefix-list-name
	Example:
	`;;Assume a preconfigured IPv6 prefix list as follows:`
	`Device(config)# ipv6 prefix-list my_prefix permit 2001:0DB8::/64 le 128`
	`;; Configure DCHPv6 Guard to match prefix`
	`Device(config-dhcp-guard)# match reply prefix-list my_prefix`
		(Optional) Enables verification of the advertised prefixes in DHCPv6 reply messages from the configured authorized prefix list. If not configured, this check will be bypassed. An empty prefix list is treated as a permit.

Step 6	[no]preference{ max limit | min limit }
	Example:
	`Device(config-dhcp-guard)# preference max 250`
	`Device(config-dhcp-guard)#preference min 150`
		Configure max and min when device-role is serverto filter DCHPv6 server advertisements by the server preference value. The defaults permit all advertisements.
		max limit—(0 to 255) (Optional) Enables verification that the advertised preference (in preference option) is less than the specified limit. Default is 255. If not specified, this check will be bypassed.
		min limit—(0 to 255) (Optional) Enables verification that the advertised preference (in

	Command or Action	Purpose
		preference option) is greater than the specified limit. Default is 0. If not specified, this check will be bypassed.
Step 7	[no] trusted-port **Example:** `Device(config-dhcp-guard)# trusted-port`	(Optional) trusted-port—Sets the port to a trusted mode. No further policing takes place on the port. **Note** If you configure a trusted port then the device-role option is not available.
Step 8	default {device-role | trusted-port} **Example:** `Device(config-dhcp-guard)# default device-role`	(Optional) default—Sets a command to its defaults.
Step 9	do show ipv6 dhcp guard policy policy_name **Example:** `Device(config-dhcp-guard)# do show ipv6 dhcp guard policy example_policy`	(Optional) Displays the configuration of the IPv6 DHCP guard policy without leaving the configuration submode. Omitting the policy_name variable displays all DHCPv6 policies.

Example of DHCPv6 Guard Configuration

```
enable
configure terminal
ipv6 access-list acl1
 permit host FE80::A8BB:CCFF:FE01:F700 any
ipv6 prefix-list abc permit 2001:0DB8::/64 le 128
ipv6 dhcp guard policy pol1
 device-role server
 match server access-list acl1
 match reply prefix-list abc
 preference min 0
 preference max 255
 trusted-port
interface GigabitEthernet 0/2/0
 switchport
 ipv6 dhcp guard attach-policy pol1 vlan add 1
 vlan 1
  ipv6 dhcp guard attach-policy pol1
show ipv6 dhcp guard policy pol1
```

How to Attach an IPv6 DHCP Guard Policy to an Interface or a VLAN on an Interface

Beginning in privileged EXEC mode, follow these steps to configure IPv6 Binding Table Content :

Procedure

	Command or Action	Purpose
Step 1	configure terminal **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 2	interface Interface_type stack/module/port **Example:** `Device(config)# interface gigabitethernet 1/1/4`	Specifies an interface type and identifier; enters the interface configuration mode.

Step 3	ipv6 dhcp guard [attach-policy policy_name [vlan {vlan_ids | add vlan_ids | except vlan_ids | none | remove vlan_ids | all}] | vlan [{vlan_ids | add vlan_ids | exceptvlan_ids | none | remove vlan_ids | all}]
	Example:
	`Device(config-if)# ipv6 dhcp guard attach-policy example_policy`
	`or`
	`Device(config-if)# ipv6 dhcp guard attach-policy example_policy vlan 222,223,224`
	`or`
	`Device(config-if)# ipv6 dhcp guard vlan 222, 223,224`
		Attaches the DHCP Guard policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.

Step 4	do show running-config interface Interface_type stack/module/port **Example:** `Device#(config-if)# do show running-config gig 1/1/4`	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 DHCP Guard Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 DHCP Guard policy on an EtherChannel interface or VLAN:

Procedure

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

	Command or Action	Purpose
	`Device# configure terminal`	
Step 2	interface range Interface_name **Example:** `Device(config)# interface Po11`	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. **Tip** Enter the do show interfaces summary command for quick reference to interface names and types.

Step 3	ipv6 dhcp guard [attach-policy policy_name [vlan {vlan_ids | add vlan_ids | except vlan_ids | none | remove vlan_ids | all}] | vlan [{vlan_ids | add vlan_ids | exceptvlan_ids | none | remove vlan_ids | all}]
	Example:
	`Device(config-if-range)# ipv6 dhcp guard attach-policy example_policy`
	`or`
	`Device(config-if-range)# ipv6 dhcp guard attach-policy example_policy vlan 222,223,224`
	`or`
	`Device(config-if-range)#ipv6 dhcp guard vlan 222, 223,224`
		Attaches the DHCP Guard policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.

Step 4	do show running-config interfaceportchannel_interface_name **Example:** `Device#(config-if-range)# do show running-config int po11`	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 DHCP Guard Policy to VLANs Globally

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 DHCP Guard policy to VLANs across multiple interfaces:

Procedure

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

	Command or Action	Purpose
	`Device# configure terminal`	
Step 2	vlan configuration vlan_list **Example:** `Device(config)# vlan configuration 334`	Specifies the VLANs to which the IPv6 Snooping policy will be attached ; enters the VLAN interface configuration mode.
Step 3	ipv6 dhcp guard [attach-policy policy_name] **Example:** `Device(config-vlan-config)#ipv6 dhcp guard attach-policy example_policy`	Attaches the IPv6 Neighbor Discovery policy to the specified VLANs across all switch and stack interfaces. The default policy is attached if the attach-policy option is not used. The default policy is, device-role client, no trusted-port.
Step 4	do show running-config **Example:** `Device#(config-if)# do show running-config`	Confirms that the policy is attached to the specified VLANs without exiting the configuration mode.

How to Configure IPv6 Source Guard

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 the global configuration mode.
Step 3	[no] ipv6 source-guard policy policy_name **Example:** `Device(config)# ipv6 source-guard policy example_policy`	Specifies the IPv6 Source Guard policy name and enters IPv6 Source Guard policy configuration mode.
Step 4	[deny global-autoconf] [permit link-local] [default{. . . }] [exit] [no{. . . }] **Example:** `Device(config-sisf-sourceguard)# deny global-autoconf`	(Optional) Defines the IPv6 Source Guard policy. • deny global-autoconf—Denies data traffic from auto-configured global addresses. This is useful when all global addresses on a link are DHCP-assigned and the

	Command or Action	Purpose
		administrator wants to block hosts with self-configured addresses to send traffic. • permit link-local—Allows all data traffic that is sourced by a link-local address. **Note** Trusted option under source guard policy is not supported.
Step 5	end **Example:** `Device(config-sisf-sourceguard)# end`	Exits out of IPv6 Source Guard policy configuration mode.
Step 6	show ipv6 source-guard policy policy_name **Example:** `Device# show ipv6 source-guard policy example_policy`	Shows the policy configuration and all the interfaces where the policy is applied.

What to do next

Apply the IPv6 Source Guard policy to an interface.

How to Attach an IPv6 Source Guard Policy to an Interface

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 the global configuration mode.
Step 3	interface Interface_type stack/module/port **Example:** `Device(config)# interface gigabitethernet 1/1/4`	Specifies an interface type and identifier; enters the interface configuration mode.
Step 4	ipv6 source-guard [attach-policy <policy_name>] **Example:**	Attaches the IPv6 Source Guard policy to the interface. The default policy is attached if the attach-policy option is not used.

	Command or Action	Purpose
	`Device(config-if)# ipv6 source-guard attach-policy example_policy`	
Step 5	show ipv6 source-guard policy policy_name **Example:** `Device#(config-if)# show ipv6 source-guard policy example_policy`	Shows the policy configuration and all the interfaces where the policy is applied.

How to attach an IPv6 Source Guard Policy to a Layer 2 EtherChannel Interface

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 the global configuration mode.
Step 3	interface port-channel port-channel-number **Example:** `Device (config)# interface Po4`	Specifies an interface type and port number and places the switch in the port channel configuration mode.
Step 4	ipv6 source-guard [attach-policy <policy_name>] **Example:** `Device(config-if) # ipv6 source-guard attach-policy example_policy`	Attaches the IPv6 Source Guard policy to the interface. The default policy is attached if the attach-policy option is not used.
Step 5	show ipv6 source-guard policy policy_name **Example:** `Device(config-if) #show ipv6 source-guard policy example_policy`	Shows the policy configuration and all the interfaces where the policy is applied.

How to Configure IPv6 Prefix Guard

Note To allow routing protocol control packets sourced by a link-local address when prefix guard is applied, enable the permit link-local command in the source-guard policy configuration mode.

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	[no] ipv6 source-guard policy source-guard-policy **Example:** `Device(config)# ipv6 source-guard policy my_snooping_policy`	Defines an IPv6 source-guard policy name and enters switch integrated security features source-guard policy configuration mode.
Step 4	[no] validate address **Example:** `Device(config-sisf-sourceguard)# no validate address`	Disables the validate address feature and enables the IPv6 prefix guard feature to be configured.
Step 5	validate prefix **Example:** `Device(config-sisf-sourceguard)# validate prefix`	Enables IPv6 source guard to perform the IPv6 prefix-guard operation.
Step 6	exit **Example:** `Device(config-sisf-sourceguard)# exit`	Exits switch integrated security features source-guard policy configuration mode and returns to privileged EXEC mode.
Step 7	show ipv6 source-guard policy [source-guard-policy] **Example:**	Displays the IPv6 source-guard policy configuration.

	Command or Action	Purpose
	`Device# show ipv6 source-guard policy policy1`	

How to Attach an IPv6 Prefix Guard Policy to an Interface

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 the global configuration mode.
Step 3	interface Interface_type stack/module/port **Example:** `Device(config)# interface gigabitethernet 1/1/4`	Specifies an interface type and identifier; enters the interface configuration mode.
Step 4	ipv6 source-guard attach-policy policy_name **Example:** `Device(config-if)# ipv6 source-guard attach-policy example_policy`	Attaches the IPv6 Source Guard policy to the interface. The default policy is attached if the attach-policy option is not used.
Step 5	show ipv6 source-guard policy policy_name **Example:** `Device(config-if)# show ipv6 source-guard policy example_policy`	Shows the policy configuration and all the interfaces where the policy is applied.

How to attach an IPv6 Prefix Guard Policy to a Layer 2 EtherChannel 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 the global configuration mode.
Step 3	interface port-channel port-channel-number **Example:** `Device (config)# interface Po4`	Specifies an interface type and port number and places the switch in the port channel configuration mode.
Step 4	ipv6 source-guard [attach-policy <policy_name>] **Example:** `Device(config-if)# ipv6 source-guard attach-policy example_policy`	Attaches the IPv6 Source Guard policy to the interface. The default policy is attached if the attach-policy option is not used.
Step 5	show ipv6 source-guard policy policy_name **Example:** `Device(config-if)# show ipv6 source-guard policy example_policy`	Shows the policy configuration and all the interfaces where the policy is applied.

Configuration Examples for IPv6 First Hop Security

Examples: How to attach an IPv6 Source Guard Policy to a Layer 2 EtherChannel Interface

The following example shows how to attach an IPv6 Source Guard Policy to a Layer 2 EtherChannel Interface:

```
Switch# configure terminal
Switch(config)# ipv6 source-guard policy POL
Switch(config-sisf-sourceguard) # validate address
switch(config-sisf-sourceguard)# exit
Switch(config)# interface Po4
Switch(config)# ipv6 snooping
Switch(config-if)# ipv6 source-guard attach-policy POL
Switch(config-if)# exit
switch(config)#
```

Examples: How to attach an IPv6 Prefix Guard Policy to a Layer 2 EtherChannel Interface

The following example shows how to attach an IPv6 Prefix Guard Policy to a Layer 2 EtherChannel Interface:

```
Switch# configure terminal
Switch(config)# ipv6 source-guard policy POL
Switch (config-sisf-sourceguard)# no validate address
Switch((config-sisf-sourceguard)# validate prefix
Switch(config)# interface Po4
```

```
Switch(config-if)# ipv6 snooping
Switch(config-if)# ipv6 source-guard attach-policy POL
```

Feature History for IPv6 First Hop Security

This table provides release and related information for features explained in this module.

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

Release	Feature	Feature Information
Cisco IOS XE Everest 16.5.1a	IPv6 First Hop Security	First Hop Security in IPv6 is a set of IPv6 security features, the policies of which can be attached to a physical interface, an EtherChannel interface, or a VLAN. An IPv6 software policy database service stores and accesses these policies. When a policy is configured or modified, the attributes of the policy are stored or updated in the software policy database, then applied as was specified. The IPv6 Snooping Policy feature has been deprecated. Although the commands are visible on the CLI and you can configure them, we recommend that you use the Switch Integrated Security Feature (SISF)-based Device Tracking feature instead. Support for this feature was introduced on all the models of the Cisco Catalyst 9500 Series Switches.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn.