Managing the Unicast RIB and FIB

This chapter describes how to manage routes in the unicast Routing Information Base (RIB) and the Forwarding Information Base (FIB) on the Cisco NX-OS device.

This chapter includes the following sections:

- About the Unicast RIB and FIB, page 14-1
- Licensing Requirements for the Unicast RIB and FIB, page 14-2
- Managing the Unicast RIB and FIB, page 14-2
- Verifying the Unicast RIB and FIB, page 14-10
- Additional References, page 14-11

About the Unicast RIB and FIB

The unicast RIB (IPv4 RIB and IPv6 RIB) and FIB are part of the Cisco NX-OS forwarding architecture, as shown in Figure 14-1.

The unicast RIB exists on the active supervisor. It maintains the routing table with directly connected routes, static routes, and routes learned from dynamic unicast routing protocols. The unicast RIB also collects adjacency information from sources such as the Address Resolution Protocol (ARP). The
unicast RIB determines the best next hop for a given route and populates the unicast forwarding information bases (FIBs) on the modules by using the services of the unicast FIB distribution module (FDM).

Each dynamic routing protocol must update the unicast RIB for any route that has timed out. The unicast RIB then deletes that route and recalculates the best next hop for that route (if an alternate path is available).

This section includes the following topics:

- **Layer 3 Consistency Checker**, page 14-2

## Layer 3 Consistency Checker

In rare instances, an inconsistency can occur between the unicast RIB and the FIB on each module. Cisco NX-OS supports the Layer 3 consistency checker. This feature detects inconsistencies between the unicast IPv4 RIB on the supervisor module and the FIB on each interface module. Inconsistencies include the following:

- Missing prefix
- Extra prefix
- Wrong next-hop address
- Incorrect Layer 2 rewrite string in the ARP or neighbor discovery (ND) cache

The Layer 3 consistency checker compares the FIB entries to the latest adjacency information from the Adjacency Manager (AM) and logs any inconsistencies. The consistency checker then compares the unicast RIB prefixes to the module FIB and logs any inconsistencies. See the “Triggering the Layer 3 Consistency Checker” section on page 14-7.

You can then manually clear any inconsistencies. See the “Clearing Forwarding Information in the FIB” section on page 14-8.

## Licensing Requirements for the Unicast RIB and FIB

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Product</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco NX-OS</td>
<td>The unicast RIB and FIB require no license. Any feature not included in a license package is bundled with the nx-os image and is provided at no extra charge to you. For a complete explanation of the Cisco NX-OS licensing scheme, see the <em>Cisco NX-OS Licensing Guide</em>.</td>
</tr>
</tbody>
</table>

## Managing the Unicast RIB and FIB

This section includes the following topics:

- **Displaying Module FIB Information**, page 14-3
- **Configuring Load Sharing in the Unicast FIB**, page 14-3
- **Displaying Routing and Adjacency Information**, page 14-5
- **Triggering the Layer 3 Consistency Checker**, page 14-7
Displaying Module FIB Information

You can display the FIB information on a module.

**DETAILED STEPS**

To display the FIB information on a module, use the following commands in any mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`show forwarding {ipv4</td>
<td>ipv6} adjacency module slot`</td>
</tr>
<tr>
<td>Example: switch# show forwarding ipv6 adjacency module 2</td>
<td></td>
</tr>
<tr>
<td>`show forwarding {ipv4</td>
<td>ipv6} route module slot`</td>
</tr>
<tr>
<td>Example: switch# show forwarding ipv6 route module 2</td>
<td></td>
</tr>
</tbody>
</table>

Configuring Load Sharing in the Unicast FIB

Dynamic routing protocols such as Open Shortest Path First (OSPF) support load balancing with equal-cost multipath (ECMP). The routing protocol determines its best routes based on the metrics configured for the protocol and installs up to the protocol-configured maximum paths in the unicast Routing Information Base (RIB). The unicast RIB compares the administrative distances of all routing protocol paths in the RIB and selects a best path set from all of the path sets installed by the routing protocols. The unicast RIB installs this best path set into the Forwarding Information Base (FIB) for use by the forwarding plane.

The forwarding plane uses a load-sharing algorithm to select one of the installed paths in the FIB to use for a given data packet.

**Note** Load sharing uses the same path for all packets in a given flow. A flow is defined by the load-sharing method that you configure. For example, if you configure source-destination load sharing, then all packets with the same source IP address and destination IP address pair follow the same path.
To configure the unicast FIB load-sharing algorithm, use the following command in global configuration mode:

```
switch(config)# ip load-sharing address source-destination
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`ip load-sharing address {destination</td>
<td>source-destination</td>
</tr>
<tr>
<td>`{ununiversal-id seed</td>
<td>rotate rotate</td>
</tr>
<tr>
<td></td>
<td>• The <code>rotate</code> option causes the hash algorithm to rotate the link picking selection so that it does not continually choose the same link across all nodes in the network. It does so by influencing the bit pattern for the hash algorithm. This option shifts the flow from one link to another and load balances the already load-balanced (polarized) traffic from the first ECMP level across multiple links. If you specify a <code>rotate</code> value, the 64-bit stream is interpreted starting from that bit position in a cyclic rotation. The <code>rotate</code> range is from 1 to 63, and the default is 32.</td>
</tr>
<tr>
<td></td>
<td>• The <code>concatenation</code> option ties together the hash tag values for ECMP and the hash tag values for port channels in order to use a stronger 64-bit hash. If you do not use this option, you can control ECMP load-balancing and port-channel load-balancing independently. The default is disabled.</td>
</tr>
</tbody>
</table>

Note: With multi-tier Layer 3 topology, polarization is possible. To avoid polarization, use a different rotate bit at each tier of the topology.

Note: To configure a rotation value for port channels, use the `port-channel load-balance src-dst ip-l4port rotate rotate` command. For more information on this command, see the *Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide*.

To display the unicast FIB load-sharing algorithm, use the following command in any mode:
To display the route that the unicast RIB and FIB use for a particular source address and destination address, use the following command in any mode:

```
switch# show routing hash 10.0.0.5 192.0.0.2
```

This example shows how to display the route selected for a source/destination pair:

```
switch# show routing hash 10.0.0.5 192.0.0.2
```

### Displaying Routing and Adjacency Information

You can display the routing and adjacency information.

To display the routing and adjacency information, use the following commands in any mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show (ip</td>
<td>Displays the unicast route table. The route-type argument can be a single route prefix, direct, static, or a dynamic route protocol. Use the ? command to see the supported interfaces.</td>
</tr>
</tbody>
</table>
| ipv6) route [route-type | interface int-type number | next-hop] | }

Example:
```
switch# show ip route
```
Chapter 14  Managing the Unicast RIB and FIB

This example shows how to display the unicast route table:

switch# show ip route
IP Route Table for Context "default"
'*' denotes best uncast next-hop       '**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
0.0.0.0/0, 1 ucast next-hops, 0 mcast next-hops
  *via 10.1.1.1, mgmt0, [1/0], 5d21h, static
0.0.0.0/32, 1 ucast next-hops, 0 mcast next-hops
  *via Null0, [220/0], 1w6d, local, discard
10.1.0.0/22, 1 ucast next-hops, 0 mcast next-hops, attached
  *via 10.1.1.55, mgmt0, [0/0], 5d21h, direct
10.1.0.0/32, 1 ucast next-hops, 0 mcast next-hops, attached
  *via 10.1.0.0, Null0, [0/0], 5d21h, local
10.1.1.0/24, 1 ucast next-hops, 0 mcast next-hops, attached
  *via 10.1.1.1, mgmt0, [2/0], 5d16h, am
10.1.1.55/32, 1 ucast next-hops, 0 mcast next-hops, attached
  *via 10.1.1.55, mgmt0, [0/0], 5d21h, local
10.1.253/32, 1 ucast next-hops, 0 mcast next-hops, attached
  *via 10.1.1.253, mgmt0, [2/0], 5d20h, am
10.1.3.255/32, 1 ucast next-hops, 0 mcast next-hops, attached
  *via 10.1.3.255, mgmt0, [0/0], 5d21h, local
255.255.255.255/32, 1 ucast next-hops, 0 mcast next-hops
  *via Eth Inband Port, [0/0], 1w6d, local

This example shows how to display the adjacency information:

switch# show ip adjacency
IP Adjacency Table for context default
Total number of entries: 2
Address   Age           MAC Address   Pref Source Interface  Best
10.1.1.1   02:20:54     00e0.b06a.71eb  50   arp       mgmt0 Yes
10.1.1.253 00:06:27     0014.5e0b.81d1  50   arp       mgmt0 Yes

This example shows how to display the adjacency information:

switch# show ip adjacency
IP Adjacency Table for context default
Total number of entries: 2
Address   Age           MAC Address   Pref Source Interface  Best
10.1.1.1   02:20:54     00e0.b06a.71eb  50   arp       mgmt0 Yes
10.1.1.253 00:06:27     0014.5e0b.81d1  50   arp       mgmt0 Yes

This example shows how to display the adjacency table. The argument ranges are as follows:
- prefix—Any IPv4 or IPv6 prefix address.
- interface-type number—Use the ? command to see the supported interfaces.
- vrf-id—Any case-sensitive, alphanumeric string up to 64 characters.

This example shows how to display the unicast route table. The route-type argument can be a single route prefix, direct, static, or a dynamic route protocol. Use the ? command to see the supported interfaces.
### Triggering the Layer 3 Consistency Checker

You can manually trigger the Layer 3 consistency checker.

To manually trigger the Layer 3 consistency checker, use the following commands in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`test forwarding [ipv4</td>
<td>ipv6] [unicast] inconsistency [vrf vrf-name] [module {slot</td>
</tr>
</tbody>
</table>

#### Example:

```
switch(config)# test forwarding inconsistency
```

To stop the Layer 3 consistency checker, use the following commands in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`test forwarding [ipv4</td>
<td>ipv6] [unicast] inconsistency [vrf vrf-name] [module {slot</td>
</tr>
</tbody>
</table>

#### Example:

```
switch# test forwarding inconsistency stop
```

To display the Layer 3 inconsistencies, use the following commands in any mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`show forwarding [ipv4</td>
<td>ipv6] inconsistency [vrf vrf-name] [module {slot</td>
</tr>
</tbody>
</table>

#### Example:

```
switch# show forwarding inconsistency
```
Clearing Forwarding Information in the FIB

You can clear one or more entries in the FIB. Clearing a FIB entry does not affect the unicast RIB.

⚠️ **Caution**

The `clear forwarding` command disrupts forwarding on the device.

To clear an entry in the FIB, including a Layer 3 inconsistency, use the following command in any mode:

```
clear forwarding (ipv4 | ipv6) route
(* | prefix) [vrf vrf-name] module
(slot | all)
```

Example:
```
switch# clear forwarding ipv4 route *
```

The `clear forwarding` command clears one or more entries from the FIB. The route options are as follows:

- `*`—All routes.
- `prefix`—Any IP or IPv6 prefix.

The `vrf-name` can be any case-sensitive, alphanumeric string up to 64 characters. The `slot` range is from 1 to 26.

Configuring Maximum Routes for the Unicast RIB

You can configure the maximum number of routes allowed in the routing table.

**SUMMARY STEPS**

1. `configure terminal`
2. `vrf context vrf-name`
3. `ipv4 unicast`
4. `maximum routes max-routes [threshold [reinstall threshold] | warning-only]`
5. (Optional) `copy running-config startup-config`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> <code>configure terminal</code></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example: switch# configure terminal switch(config)#</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> <code>vrf context vrf-name</code></td>
<td>Creates a VRF and enters VRF configuration mode.</td>
</tr>
<tr>
<td>Example: switch(config)# vrf context Red switch(config-vrf)#</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> <code>ipv4 unicast</code></td>
<td>Enters address-family configuration mode.</td>
</tr>
<tr>
<td>Example: switch(config-vrf)# ipv4 unicast switch(config-vrf-af-ipv4)#</td>
<td></td>
</tr>
</tbody>
</table>
Chapter 14  Managing the Unicast RIB and FIB

Managing the Unicast RIB and FIB

Estimating Memory Requirements for Routes

You can estimate the memory that a number of routes and next-hop addresses will use.

To estimate the memory requirements for routes, use the following command in any mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show routing {ipv6} memory estimate routes num-routes next-hops num-nexthops</td>
<td>Displays the memory requirements for routes. The num-routes range is from 1000 to 1000000. The num-nexthops range is from 1 to 16.</td>
</tr>
</tbody>
</table>

Example:

switch# show routing memory estimate routes 5000 next-hops 2

Clearing Routes in the Unicast RIB

You can clear one or more routes from the unicast RIB.

Caution

The * keyword is severely disruptive to routing.

To clear one or more entries in the unicast RIB, use the following commands in any mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>maximum routes max-routes [threshold [reinstall threshold]</td>
<td>warning-only]</td>
</tr>
<tr>
<td>threshold—Percentage of maximum routes that triggers a warning message. The range is from 1 to 100.</td>
<td></td>
</tr>
<tr>
<td>warning-only—Logs a warning message when the maximum number of routes is exceeded.</td>
<td></td>
</tr>
<tr>
<td>reinstall threshold—Reinstalls routes that previously exceeded the maximum route limit and were rejected and specifies the threshold value at which to reinstall them. The threshold range is from 1 to 100.</td>
<td></td>
</tr>
</tbody>
</table>

Example:

switch(config-vrf-af-ipv4)# maximum routes 250 90

Step 5

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>copy running-config startup-config</td>
<td>(Optional) Saves this configuration change.</td>
</tr>
</tbody>
</table>

Example:

switch(config-vrf-af-ipv4)# copy running-config startup-config
Verifying the Unicast RIB and FIB

To display the unicast RIB and FIB information, perform one of the following tasks:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>clear {ip</td>
<td>ipv4</td>
</tr>
<tr>
<td>clear routing {multicast</td>
<td>unicast} {ip</td>
</tr>
</tbody>
</table>

Example:

```
switch(config)# clear ip route 10.2.2.2
```

```
switch(config)# clear routing ip 10.2.2.2
```
Additional References

For additional information related to managing unicast RIB and FIB, see the following sections:

- Related Documents, page 14-11

Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuring EEM</td>
<td><em>Cisco Nexus 9000 Series NX-OS System Management Configuration Guide</em></td>
</tr>
</tbody>
</table>