Configuring IP Multicast over Unidirectional Links

Last Updated: October 16, 2012

IP multicast requires bidirectional communication, yet some networks include broadcast satellite links, which are unidirectional. Unidirectional link routing (UDLR) provides three mechanisms for a router to emulate a bidirectional link to enable the routing of unicast and multicast packets over a physical unidirectional interface, such as a broadcast satellite link. The mechanisms are a UDLR tunnel, Internet Group Management Protocol (IGMP) UDLR, and IGMP proxy. This document describes a UDLR tunnel and IGMP UDLR. IGMP proxy is described in the “Customizing IGMP” module. The three mechanisms may be used independently or in combination.

• Finding Feature Information, page 1
• Prerequisites for UDLR, page 1
• Information About UDLR, page 2
• How to Route IP Multicast over Unidirectional Links, page 3
• Configuration Examples for UDLR, page 9
• Additional References, page 14
• Feature Information for Configuring IP Multicast over Unidirectional Links, page 15

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for UDLR

• You understand the concepts in the “IP Multicast Technology Overview” module.
You have IP multicast configured in your network. Refer to the “Configuring Basic IP Multicast” module.

**Information About UDLR**

- UDLR Overview, page 2
- UDLR Tunnel, page 2
- IGMP UDLR, page 3

**UDLR Overview**

Both unicast and multicast routing protocols forward data on interfaces from which they have received routing control information. This model requires a bidirectional link. However, some network links are unidirectional. For networks that are unidirectional (such as broadcast satellite links), a method of communication that allows for control information to operate in a unidirectional environment is necessary. (Note that IGMP is not a routing protocol.)

Specifically, in unicast routing, when a router receives an update message on an interface for a prefix, it forwards data for destinations that match that prefix out that same interface. This is the case in distance vector routing protocols. Similarly, in multicast routing, when a router receives a Join message for a multicast group on an interface, it forwards copies of data destined for that group out that same interface. Based on these principles, unicast and multicast routing protocols cannot be supported over UDLs without the use of UDLR. UDLR is designed to enable the operation of routing protocols over UDLs without changing the routing protocols themselves.

UDLR enables a router to emulate the behavior of a bidirectional link for IP operations over UDLs. UDLR has three complementary mechanisms for bidirectional link emulation, which are described in the following sections:

- UDLR Tunnel--A mechanism for routing unicast and multicast traffic.
- Internet Group Management Protocol (IGMP) UDLR--Mechanism for routing multicast traffic. This method scales well for many broadcast satellite links.
- IGMP Proxy--Mechanism for routing multicast traffic.

You can use each mechanism independently or in conjunction with the others. IGMP proxy is described in the “Customizing IGMP” module.

**UDLR Tunnel**

The UDLR tunnel mechanism enables IP and its associated unicast and multicast routing protocols to treat the unidirectional link (UDL) as being logically bidirectional. A packet that is destined on a receive-only interface is picked up by the UDLR tunnel mechanism and sent to an upstream router using a generic routing encapsulation (GRE) tunnel. The control traffic flows in the opposite direction of the user data flow. When the upstream router receives this packet, the UDLR tunnel mechanism makes it appear that the packet was received on a send-only interface on the UDL.

The purpose of the unidirectional GRE tunnel is to move control packets from a downstream node to an upstream node. The one-way tunnel is mapped to a one-way interface (that goes in the opposite direction). Mapping is performed at the link layer, so the one-way interface appears bidirectional. When the upstream node receives packets over the tunnel, it must make the upper-layer protocols act as if the packets were received on the send-capable UDL.
A UDLR tunnel supports the following functionality:

- Address Resolution Protocol (ARP) and Next Hop Resolution Protocol (NHRP) over a UDL
- Emulation of bidirectional links for all IP traffic (as opposed to only control-only broadcast/multicast traffic)
- Support for IP GRE multipoint at a receive-only tunnel

Note

A UDL router can have many routing peers (for example, routers interconnected via a broadcast satellite link). As with bidirectional links, the number of peer routers a router has must be kept relatively small to limit the volume of routing updates that must be processed. For multicast operation, we recommend using the IGMP UDLR mechanism when interconnecting more than 20 routers.

IGMP UDLR

In addition to a UDLR tunnel, another mechanism that enables support of multicast routing protocols over UDLs is using IP multicast routing with IGMP, which accommodates UDLR. This mechanism scales well for many broadcast satellite links.

With IGMP UDLR, an upstream router sends periodic queries for members on the UDL. The queries include a unicast address of the router that is not the unicast address of the unidirectional interface. The downstream routers forward IGMP reports received from directly connected members (on interfaces configured to helper forward IGMP reports) to the upstream router. The upstream router adds the unidirectional interface to the ( *, G) outgoing interface list, thereby enabling multicast packets to be forwarded down the UDL.

In a large enterprise network, it is not possible to receive IP multicast traffic via satellite and forward the traffic throughout the network. This limitation exists because receiving hosts must be directly connected to the downstream router. However, you can use the IGMP proxy mechanism to overcome this limitation. Refer to the “ Customizing IGMP ” module for more information on this mechanism.

How to Route IP Multicast over Unidirectional Links

This section includes the following procedures. You can do either or both in your network.

- Configuring a UDLR Tunnel, page 3
- Configuring IGMP UDLR, page 6

Configuring a UDLR Tunnel

To configure a UDLR tunnel, perform the task in this section. The tunnel mode defaults to GRE. You need not assign an IP address to the tunnel (you need not use the `ip address` or `ip unnumbered` commands). You must configure the tunnel endpoint addresses.

You must configure both the upstream and downstream routers to meet the following conditions:

- On the upstream router, where the UDL can only send, you must configure the tunnel to receive. When packets are received over the tunnel, the upper-layer protocols treat the packet as though it is received over the unidirectional, send-only interface.
• On the downstream router, where the UDL can only receive, you must configure the tunnel to send. When packets are sent by upper-layer protocols over the interface, they will be redirected and sent over this GRE tunnel.

Before configuring UDLR tunnel, ensure that all routers on the UDL have the same subnet address. If all routers on the UDL cannot have the same subnet address, the upstream router must be configured with secondary addresses to match all the subnets that the downstream routers are attached to.

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `interface type number`
4. `interface tunnel number`
5. `tunnel udlr receive-only type number`
6. `tunnel source {ip-address | type number}`
7. `tunnel destination {hostname | ip-address}`
8. Move to the downstream router.
9. `enable`
10. `configure terminal`
11. `interface type number`
12. `interface tunnel number`
13. `tunnel udlr send-only type number`
14. `tunnel source {ip-address | type number}`
15. `tunnel destination {hostname | ip-address}`
16. `tunnel udlr address-resolution`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router&gt; enable</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td></td>
<td>• Do this step on the upstream router.</td>
</tr>
<tr>
<td>Step 2 configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
</tr>
<tr>
<td><strong>Step 3</strong> interface type number</td>
<td>Configures the unidirectional send-only interface.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router(config)# interface gigabitethernet 0/0/0</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> interface tunnel number</td>
<td>Configures the receive-only tunnel interface.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router(config-if)# interface tunnel 0</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> tunnel udlr receive-only type number</td>
<td>Configures the UDLR tunnel.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router(config-if)# tunnel udlr receive-only fastethernet 0/0/0</td>
<td></td>
</tr>
<tr>
<td>• Use the same <code>type</code> and <code>number</code> values as the unidirectional send-only interface <code>type</code> and <code>number</code> values specified with the <code>interface type number</code> command in Step 3.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong> tunnel source {ip-address</td>
<td>type number}</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router(config-if)# tunnel source 10.3.4.5</td>
<td></td>
</tr>
<tr>
<td><strong>Step 7</strong> tunnel destination {hostname</td>
<td>ip-address}</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router(config-if)# tunnel destination 10.8.2.3</td>
<td></td>
</tr>
<tr>
<td><strong>Step 8</strong> Move to the downstream router.</td>
<td></td>
</tr>
<tr>
<td>--</td>
<td></td>
</tr>
<tr>
<td><strong>Step 9</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router&gt; enable</td>
<td></td>
</tr>
<tr>
<td>• Enter your password if prompted.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 10</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router# configure terminal</td>
<td></td>
</tr>
</tbody>
</table>
### Step 11: `interface type number`

**Example:**

```
Router(config)# interface gigabitethernet 0/0/0
```

**Purpose:** Configures the unidirectional receive-only interface.

### Step 12: `interface tunnel number`

**Example:**

```
Router(config-if)# interface tunnel 0
```

**Purpose:** Configures the send-only tunnel interface.

### Step 13: `tunnel udlr send-only type number`

**Example:**

```
Router(config-if)# tunnel udlr send-only ethernet 0
```

**Note:** Use the same `type` and `number` values as the unidirectional receive-only interface `type` and `number` values specified with the `interface type number` command in Step 3.

### Step 14: `tunnel source {ip-address | type number}`

**Example:**

```
Router(config-if)# tunnel source 11.8.2.3
```

**Purpose:** Configures the tunnel source.

### Step 15: `tunnel destination {hostname | ip-address}`

**Example:**

```
Router(config-if)# tunnel destination 10.3.4.5
```

**Purpose:** Configures the tunnel destination.

### Step 16: `tunnel udlr address-resolution`

**Example:**

```
Router(config-if)# tunnel udlr address-resolution
```

**Purpose:** Enables the forwarding of ARP and NHRP.

---

## Configuring IGMP UDLR

To configure an IGMP UDL, you must configure both the upstream and downstream routers. You need not specify whether the direction is sending or receiving; IGMP learns the direction by the nature of the physical connection.

When the downstream router receives an IGMP report from a host, the router sends the report to the IGMP querier associated with the UDL interface identified in the `ip igmp helper-address` command.
• All routers on the UDL have the same subnet address. If all routers on the UDL cannot have the same subnet address, the upstream router must be configured with secondary addresses to match all the subnets that the downstream routers are attached to.
• Multicast receivers are directly connected to the downstream routers.

**SUMMARY STEPS**

1. enable
2. configure terminal
3. interface type number
4. ip igmp unidirectional-link
5. Move to the downstream router.
6. enable
7. configure terminal
8. ip multicast default-rpf-distance distance
9. interface type number
10. ip igmp unidirectional-link
11. ip igmp helper-address udl type number
12. exit
13. show ip igmp udlr [group-name| group-address | type number]

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td></td>
<td>• Begin on the upstream router.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router&gt; enable</td>
<td></td>
</tr>
<tr>
<td>Step 2 configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td>Step 3 interface type number</td>
<td>Configures the interface.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router(config)# interface</td>
<td></td>
</tr>
<tr>
<td>gigabitethernet 0/1/1</td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
</tr>
<tr>
<td>Step 4 ip igmp unidirectional-link</td>
<td>Configures IGMP on the interface to be unidirectional.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router(config-if)# ip igmp unidirectional-link</td>
<td></td>
</tr>
<tr>
<td>Step 5 Move to the downstream router.</td>
<td>--</td>
</tr>
<tr>
<td>Step 6 enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router&gt; enable</td>
<td></td>
</tr>
<tr>
<td>Step 7 configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td>Step 8 ip multicast default-rpf-distance distance</td>
<td>(Optional) Sets the distance for the default RPF interface.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router# ip multicast default-rpf-distance 10</td>
<td>By default, the distance for the default reverse path forwarding (RPF) interface is 15. Any explicit sources learned by routing protocols will take preference if their distance is less than the distance configured by the <code>ip multicast default-rpf-distance</code> command. Use this command on downstream routers if you want some sources to use RPF to reach the UDLR link and others to use the terrestrial paths.</td>
</tr>
<tr>
<td></td>
<td>• If you want IGMP to prefer the UDL, set the distance to be less than the distances of the unicast routing protocols.</td>
</tr>
<tr>
<td></td>
<td>• If you want IGMP to prefer the non-UDL, set the distance to be greater than the distances of the unicast routing protocols.</td>
</tr>
<tr>
<td>Step 9 interface type number</td>
<td>Configures the interface.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router(config)# interface gigabitethernet 0/0/0</td>
<td></td>
</tr>
<tr>
<td>Step 10 ip igmp unidirectional-link</td>
<td>Configures IGMP on the interface to be unidirectional.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router(config-if)# ip igmp unidirectional-link</td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
</tr>
<tr>
<td><strong>Step 11</strong> ip igmp helper-address udl <em>type number</em></td>
<td>Configures the interface to be an IGMP helper.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td><strong>Step 12</strong> exit</td>
<td>Exits configuration mode and returns to EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td><strong>Step 13</strong> show ip igmp udlr [group-name</td>
<td>group-address</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
</tbody>
</table>

### Configuration Examples for UDLR

- UDLR Tunnel Example, page 9
- IGMP UDLR Example, page 11
- Integrated UDLR Tunnel IGMP UDLR and IGMP Proxy Example, page 12

### UDLR Tunnel Example

The following example shows how to configure a UDLR tunnel. In the example, Router A (the upstream router) is configured with Open Shortest Path First (OSPF) and PIM. Serial interface 0 has send-only capability. Therefore, the UDLR tunnel is configured as receive only, and points to serial 0.
Router B (the downstream router) is configured with OSPF and PIM. Serial interface 1 has receive-only capability. Therefore, the UDLR tunnel is configured as send-only, and points to serial 1. The forwarding of ARP and NHRP is enabled. The figure below illustrates the example.

**Figure 1  UDLR Tunnel Example**

![UDLR Tunnel Example](image)

**Router A Configuration**

```plaintext
ip multicast-routing
!
! Serial0/0/0 has send-only capability
!
interface serial 0/0/0
  encapsulation hdlc
  ip address 10.1.0.1 255.255.0.0
  ip pim sparse-dense-mode
!
! Configure tunnel as receive-only UDLR tunnel.
!
interface tunnel 0
  tunnel source 10.20.0.1
  tunnel destination 10.41.0.2
  tunnel udlr receive-only serial 0/0/0
!
! Configure OSPF.
!
router ospf
  network 10.0.0.0 0.255.255.255 area 0
```

**Router B Configuration**

```plaintext
ip multicast-routing
!
! Serial1 has receive-only capability
!
interface serial 1/0/0
  encapsulation hdlc
  ip address 10.1.0.2 255.255.0.0
  ip pim sparse-dense-mode
!
! Configure tunnel as send-only UDLR tunnel.
!
interface tunnel 0
  tunnel source 10.41.0.2
```
tunnel destination 10.20.0.1
tunnel udlr send-only serial 1/0/0
tunnel udlr address-resolution
!
! Configure OSPF.
!
router ospf
    network 10.0.0.0 0.255.255.255 area 0

IGMP UDLR Example

The following example shows how to configure IGMP UDLR. In this example, uplink-rtr is the local upstream router and downlink-rtr is the downstream router.

Both routers are also connected to each other by a back channel connection. Both routers have two IP addresses: one on the UDL and one on the interface that leads to the back channel. The back channel is any return route and can have any number of routers.

Note
Configuring PIM on the back channel interfaces on the uplink router and downlink router is optional.

All routers on a UDL must have the same subnet address. If all routers on a UDL cannot have the same subnet address, the upstream router must be configured with secondary addresses to match all the subnets that the downstream routers are attached to.

Figure 2  IGMP Unidirectional Link Routing Example

Uplink Router (uplink-rtr) Configuration

ip multicast-routing
Integrated UDLR Tunnel IGMP UDLR and IGMP Proxy Example

The following example shows how to configure UDLR tunnels, IGMP UDLR, and IGMP proxy on both the upstream and downstream routers sharing a UDL.

**Upstream Configuration**

```conf
ip multicast-routing

! Interface that source is attached to
interface gigabitethernet 0/0/0
description Typical IP multicast enabled interface
ip address 12.0.0.1 255.0.0.0
ip pim sparse-dense-mode

! Back channel
interface gigabitethernet 1/0/0
description Back channel which has connectivity to downlink-rtr
ip address 11.0.0.1 255.0.0.0
ip pim sparse-dense-mode

! Unidirectional link
interface serial 0/0/0
description Unidirectional to downlink-rtr
ip address 10.0.0.1 255.0.0.0
ip pim sparse-dense-mode
ip igmp unidirectional-link
no keepalive
```

**Downlink Router (downlink-rtr) Configuration**

```conf
ip multicast-routing

! Interface that receiver is attached to, configure for IGMP reports to be helpered for the unidirectional interface.
interface gigabitethernet 0/0/0
description Typical IP multicast-enabled interface
ip address 14.0.0.2 255.0.0.0
ip pim sparse-dense-mode
ip igmp helper-address udl serial 0/0/0

! Back channel
interface gigabitethernet 1/0/0
description Back channel that has connectivity to downlink-rtr
ip address 13.0.0.2 255.0.0.0
ip pim sparse-dense-mode

! Unidirectional link
interface serial 0/0/0
description Unidirectional to uplink-rtr
ip address 10.0.0.2 255.0.0.0
ip pim sparse-dense-mode
ip igmp unidirectional-link
no keepalive
```
tunnel udlr receive-only GigabitEthernet2/3/0

interface GigabitEthernet2/0/0
  no ip address
  shutdown

! user network
interface GigabitEthernet2/1/0
  ip address 9.1.89.1 255.255.255.240
  no ip directed-broadcast
  ip pim dense-mode
  ip cgmp
  fair-queue 64 256 128
  no cdp enable
  ip rsvp bandwidth 1000 100

interface GigabitEthernet2/2/0
  ip address 9.1.95.1 255.255.255.240
  no ip directed-broadcast

! physical send-only interface
interface GigabitEthernet2/3/0
  ip address 9.1.92.100 255.255.255.240
  no ip directed-broadcast
  ip pim dense-mode
  ip nhrp network-id 5
  ip nhrp server-only
  ip igmp unidirectional-link
  fair-queue 64 256 31
  ip rsvp bandwidth 1000 100

router ospf 1
  network 9.1.92.96 0.0.0.15 area 1

! ip classless
ip route 9.1.90.0 255.255.255.0 9.1.92.99

**Downstream Configuration**

ip multicast-routing

! interface Loopback0
  ip address 9.1.90.161 255.255.255.252
  ip pim sparse-mode
  ip igmp helper-address udl GigabitEthernet2/3/0
  ip igmp proxy-service

! interface Tunnel0
  ip address 9.1.90.97 255.255.255.252
  ip access-group 120 out
  no ip directed-broadcast
  no ip mroute-cache
tunnel source 9.1.90.97
tunnel destination 9.1.89.97
tunnel key 5
tunnel udlr send-only GigabitEthernet2/3/0
tunnel udlr address-resolution

! interface GigabitEthernet2/0/0
  no ip address
  no ip directed-broadcast
  shutdown
  no cdp enable

! user network
interface GigabitEthernet2/1/0
  ip address 9.1.90.1 255.255.255.240
  no ip directed-broadcast
  ip pim sparse-mode
  ip igmp mroute-proxy Loopback0
  no cdp enable

!
! Backchannel
interface GigabitEthernet2/2/0
  ip address 9.1.95.3 255.255.255.240
  no ip directed-broadcast
  no cdp enable
!
! physical receive-only interface
interface GigabitEthernet2/3/0
  ip address 9.1.92.99 255.255.255.240
  no ip directed-broadcast
  ip pim sparse-mode
  ip igmp unidirectional-link
  no keepalive
  no cdp enable
!
router ospf 1
  network 9.1.90.0 0.0.0.255 area 1
  network 9.1.92.96 0.0.0.15 area 1
!
ip classless
ip route 0.0.0.0 0.0.0.0 9.1.95.1
!
! set rpf to be the physical receive-only interface
ip mr route 0.0.0.0 0.0.0.0 9.1.92.96
ip pim rp-address 9.1.90.1
!
! permit ospf, ping and rsvp, deny others
access-list 120 permit icmp any any
access-list 120 permit 46 any any
access-list 120 permit ospf any any

Additional References

Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP multicast commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples</td>
<td>Cisco IOS IP Multicast Command Reference</td>
</tr>
<tr>
<td>Tunnel interfaces</td>
<td>“Implementing Tunnels” module</td>
</tr>
<tr>
<td>IGMP and IGMP Proxy</td>
<td>“Customizing IGMP” module</td>
</tr>
</tbody>
</table>

MIBs

<table>
<thead>
<tr>
<th>MIB</th>
<th>MIBs Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>To locate and download MIBs for selected platforms, Cisco IOS XE releases, and feature sets, use Cisco MIB Locator found at the following URL:</td>
</tr>
<tr>
<td></td>
<td><a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a></td>
</tr>
</tbody>
</table>
Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/techsupport">http://www.cisco.com/techsupport</a></td>
</tr>
</tbody>
</table>

Feature Information for Configuring IP Multicast over Unidirectional Links

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Configuration Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>UDLR Tunnel ARP and IGMP Proxy</td>
<td>12.2(8)T</td>
<td>This feature enables arp over a unidirectional link and overcomes the existing limitation of requiring downstream multicast receivers to be directly connected to the unidirectional link downstream router.</td>
</tr>
<tr>
<td>Feature Name</td>
<td>Releases</td>
<td>Feature Configuration Information</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>-------------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Uni-Directional Link Routing (UDLR)</td>
<td>12.2(2)T</td>
<td>Unidirectional link routing is used to allow routing protocols to function in environments where routers are connected through unidirectional links.</td>
</tr>
<tr>
<td></td>
<td>12.2(17d)SXB1</td>
<td>Unidirectional link routing enables layer 3 connectivity by tunneling routing information to the router on the upstream side of a unidirectional link.</td>
</tr>
</tbody>
</table>