Chapter 21

High Availability

This chapter covers the following topics:

- Continuous forwarding
- Failure avoidance
- Event-driven failure detection
- Fast routing convergence
- Fast reroute

In networking, *availability* refers to the operational uptime of the network. The aim of *high availability* is to achieve continuous network uptime by designing a network to avoid single points of failure, incorporate deterministic network patterns, and utilize event-driven failure detection to provide fast network convergence. This chapter outlines various techniques for improving availability by reducing packet loss and accelerating network convergence around failures.

> **Note** The features discussed in this chapter are not universally available for all Cisco router platforms. Use the software feature navigator tool on Cisco's website http://www.cisco.com to verify your minimum hardware and software requirements.
>
> For demonstration purposes, the configuration examples in the text use the IPv4 address family, yet in most cases the features also apply to IPv6.

Network Convergence Overview

Network convergence is the time required to redirect traffic around a failure that causes loss of connectivity (LoC). The latency requirements for convergence vary by application. For example, an Open Shortest Path First (OSPF) network using default settings may take 5 or more seconds to converge around a link failure. This length of time may be acceptable for users reading Internet website articles, but it is completely unacceptable for IP telephony users.

A number of factors influence network convergence speed. In general, the higher the number of network prefixes and routers in the network, the slower the convergence will be. The primary factors that influence network convergence are as follows:

- **T1:** Time to **detect** the failure event
- **T2:** Time to **propagate** the event to neighbors
- **T3:** Time to **process** the event and calculate new best path
- **T4:** Time to **update** the routing table and program forwarding tables

Figure 21-1 illustrates the various time intervals that influence network convergence.

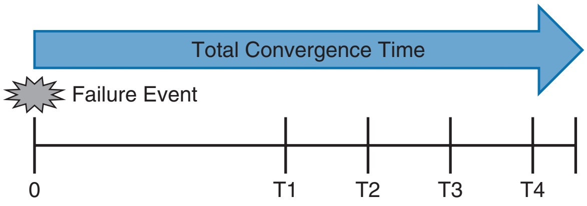

Figure 21-1 *Convergence Time*

Continuous Forwarding

Routers specifically designed for high availability include hardware redundancy, such as dual power supplies and route processors (RPs). An RP, which is also called a *supervisor* on some platforms, is responsible for learning the network topology and building the route table (Routing Information Base [RIB]). An RP failure can trigger routing protocol adjacencies to reset, resulting in packet loss and network instability. During an RP failure, it may be more desirable to hide the failure and allow the router to continue forwarding packets using the previously programmed Cisco Express Forwarding (CEF) table entries versus temporarily dropping packets while waiting for the secondary RP to reestablish the routing protocol adjacencies and rebuild the forwarding table.

The following two high availability features allow the network to route through a failure during an RP switchover:

- Stateful switchover (SSO) with nonstop forwarding (NSF)
- Stateful switchover (SSO) with nonstop routing (NSR)

Stateful Switchover

Stateful switchover (SSO) is a redundancy feature that allows a Cisco router with two RPs to synchronize router configuration and control plane state information. The process of mirroring information between RPs is referred to as *checkpointing*. SSO-enabled routers always checkpoint line card operation and Layer 2 protocol states. During a switchover, the standby RP immediately takes control and will prevent problems such as interface link flaps and router reloads; however, Layer 3 packet forwarding is disrupted without additional configuration. The standby RP does not have any Layer 3 checkpoint information about the routing peer, so a switchover will trigger a routing protocol adjacency flap that clears the route table. After the route table is cleared, the CEF entries are purged, and traffic is no longer routed until the network topology is relearned and the forwarding table is reprogrammed. Enabling NSF or NSR high availability capabilities informs the routers to maintain the CEF entries for a short duration and continue forwarding packets through an RP failure until the control plane recovers.

SSO requires that both RPs have the same software version. Example 21-1 demonstrates enabling stateful switchover using the redundancy mode configuration command **mode sso**. The feature is automatically enabled by default on all IOS XR routers.

Example 21-1 *Supervisor SSO Redundancy*

```
IOS
redundancy
mode sso
```

The IOS and IOS XR command **show redundancy** provides details on the current SSO state operation.

Example 21-2 displays the current redundancy status for an IOS router and an IOS XR router. The primary RP is highlighted in the example. In IOS terminology, the STANDBY HOT router is the backup RP, while in IOS XR the Standby is the backup RP.

Example 21-2 *SSO Redundancy Status*

```
R1#show redundancy
! Output omitted for brevity
Current Processor Information :
-------------------------------
               Active Location = slot 5
        Current Software state = ACTIVE

Peer Processor Information :
----------------------------
              Standby Location = slot 6
        Current Software state = STANDBY HOT
```

```
RP/0/RP1/CPU0:XR3#show redundancy summary
  Active/Primary   Standby/Backup
  --------------   --------------
   0/RP1/CPU0(A)     0/RP0/CPU0(S) (Node Ready, NSR: Ready)
   0/RP1/CPU0(P)     0/RP0/CPU0(B) (Proc Group Ready, NSR: Ready)
```

Manually triggering a switchover between route processors is performed with the command **redundancy force-switchover** on IOS routers and with the command **redundancy switchover** on IOS XR nodes.

Nonstop Forwarding and Graceful Restart

Nonstop forwarding (NSF) is a feature deployed along with SSO to protect the Layer 3 forwarding plane during an RP switchover. With NSF enabled, the router continues to forward packets using the stored entries in the Forwarding Information Base (FIB) table.

There are three categories of NSF routers:

- **NSF-capable router:** A router that has dual RPs and is manually configured to use NSF to preserve the forwarding table through a switchover. The router restarts the routing process upon completion of the RP switchover.
- **NSF-aware router:** A neighbor router, which assists the NSF-capable router during the restart by preserving the routes and adjacency state during the RP switchover. An NSF-aware router does not require dual RPs.
- **NSF-unaware router:** A router that is not aware or capable of assisting a neighboring router during an RP switchover.

SSO with NSF is a high availability feature that is part of the internal router operation. Graceful restart (GR) is a subcomponent of NSF and is the mechanism the routing protocols use to signal NSF capabilities and awareness.

In Figure 21-2, R1 has NSF with SSO enabled. The primary RP has failed, but the router continues to forward packets using the existing CEF tables (FIB). During this time, the backup RP transparently takes over and reestablishes communication with R2 to restore the control plane and repopulate the routing tables (RIB). Throughout this *grace period*, R2 does not notify the rest of the network that a failure has occurred on R1, which maintains stability in the network and prevents a networkwide topology change event.

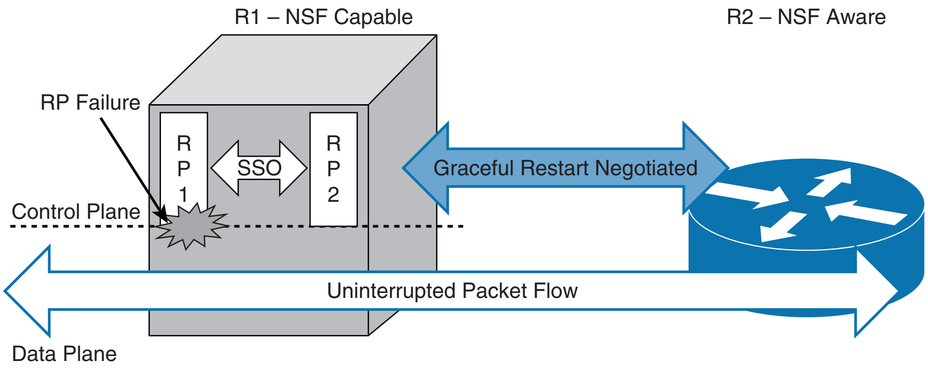

Figure 21-2 *SSO with NSF*

Table 21-1 outlines the expected routing protocol interaction between R1 and R2 when an RP switchover occurs on R1.

Table 21-1 *NSF Routing Protocol Interaction*

	SSO with NSF (R1 --> R2 Graceful Restart Signaling)			
	Neighbor Adjacency Reset		**RIB Recalculated Route Age Refreshed**	
	R1	**R2**	**R1**	**R2**
OSPF	Yes	No	Yes	No
IS-IS	Yes (IETF)	No	Yes	No
EIGRP	Yes	No	Yes	No
BGP	Yes	Yes	Yes	No

The GR signaling mechanism differs slightly for each protocol, but the general concept is the same for all. Multiple RFC standards exist describing in detail how the GR should be communicated by each routing protocol. You can find a list of the relevant RFCs in the reference section at the end of the chapter.

Figure 21-3 illustrates the NSF capability exchange:

- R1 signals to R2 that it is NSF/SSO-capable while forming the initial routing adjacency. The two agree that if R1 should signal a control plane reset, R2 will not drop the peering and will continue sending and receiving traffic to R1 as long as the routing protocol hold timers do not expire.
- R1 sends a GR message to R2 indicating that the control plane is temporarily going offline immediately preceding the RP failover.

- R1 maintains the CEF table programming to forward traffic to R2 while the RP switchover takes place.
- Upon completion of the switchover, the new primary RP on R1 reestablishes communication with R2 and requests updates for repopulating the route table.

R2 provides the route information to R1, while at the same time suppressing a notification to the rest of the network that an adjacency flap has occurred to R1. Stability is improved by preventing an unnecessary networkwide best path recalculation for the route flap.

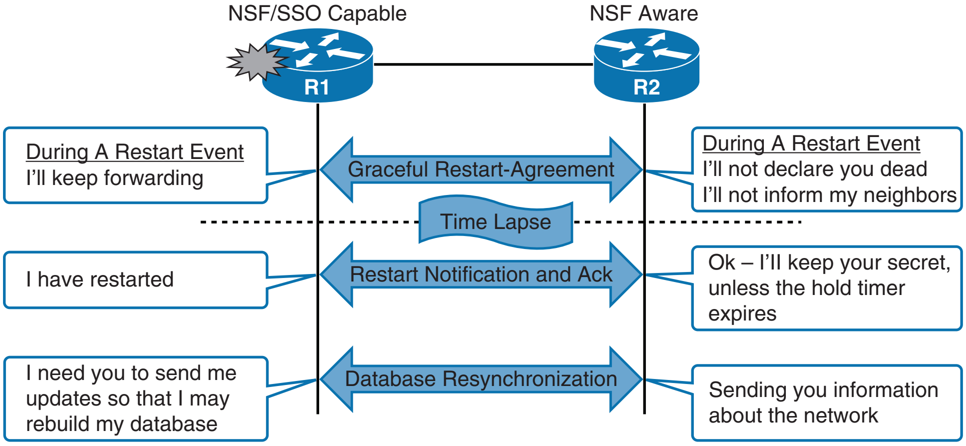
NSF/SSO Capable
NSF Aware
R1
R2
During A Restart Event
I'll keep forwarding
Graceful Restart-Agreement
During A Restart Event
I'll not declare you dead
I'll not inform my neighbors
Time Lapse
I have restarted
Restart Notification and Ack
Ok – I'll keep your secret, unless the hold timer expires
I need you to send me updates so that I may rebuild my database
Database Resynchronization
Sending you information about the network

Figure 21-3 *NSF Graceful Restart Relationship Building Process*

Note NSF *freezes* the CEF table and allows the router to forward packets to the last known good next-hop from prior to the RP switchover. If the network topology changes while the router is recovering, the packets may be suboptimally routed or possibly sent to the wrong destination and dropped.

NSF should not be deployed in parallel with routing protocol keepalive and holddown timers of less than 4 seconds. The NSF-capable router requires time to reestablish control plane communication during an RP switchover, and aggressive holddown timers can expire before this activity completes, leading to a neighbor adjacency flap. Bidirectional forwarding detection (BFD) is a better solution in most scenarios than aggressive keepalive timers. BFD is discussed in detail later in this chapter.

The interior routing protocols Open Shortest Path First (OSPF) Protocol, Intermediate System-to-Intermediate System (IS-IS) Protocol, and Enhanced Interior Gateway Routing Protocol (EIGRP) are automatically NSF-aware for both IOS and IOS XR. Routers with dual RPs need to be configured as NSF-capable within the routing protocol.

The routers in Figure 21-4 are referenced throughout the next series of continuous forwarding examples. In each of the exercises, R1 is the NSF-capable router that is performing an RP stateful switchover. Where possible, verification captures from the NSF-aware routers R2 and XR3 demonstrate the GR signaling.

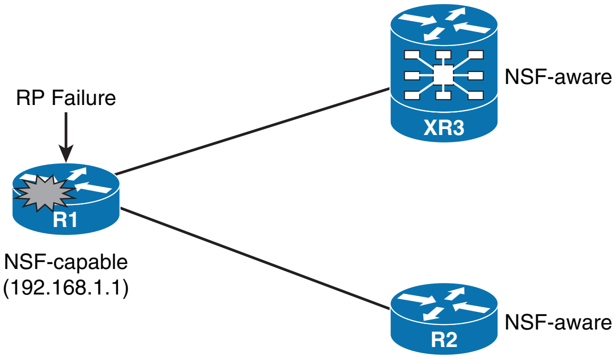

Figure 21-4 *NSF Routers*

OSPF

Two GR configuration modes are available for OSPF:

- **Cisco:** The Cisco mode for performing GR adds Link Local Signaling (LLS) bits to the hello and DBD packets. The LSDB Resynchronization (LR) bit is included in the database description (DBD) packets to indicate out-of-band resynchronization (OOB) capability. A hello packet with the LLS Restarting R bit set indicates that the router is about to perform a restart. This method was developed by Cisco and was later standardized by the IETF in RFC 4811, 4812, and 5613.
- **IETF:** The IETF RFC 3623 method for performing GR uses link-local opaque link-state advertisements (LSAs). A router sends a Grace LSA to indicate it is about to restart the OSPF process. The router resynchronizes the LSDB using Grace LSAs once the restart completes.

The routing protocol command to enable GR and NSF capabilities is **nsf** [**cisco** | **ietf**]. Example 21-3 demonstrates how to enable a router to be NSF-capable and to signal GR capabilities to its peers.

Example 21-3 *OSPF NSF Configuration*

```
IOS
router ospf 100
nsf cisco
```

```
IOS XR
router ospf 100
 nsf cisco
```

You can view the NSF status with the IOS command **show ip ospf neighbor detail** or with the IOS XR command **show ospf neighbor detail**.

Example 21-4 demonstrates that the neighbors are using LLS, which is required for NSF awareness and successful GR negotiations. Notice that the neighbor adjacency peering includes the LLS LSDB OOB Resynchronization (LR) capability bit set.

Example 21-4 *NSF Graceful Restart Agreement*

```
R2#show ip ospf neighbor detail
! Output omitted for brevity
 Neighbor 192.168.1.1, interface address 10.0.12.1
    In the area 0 via interface GigabitEthernet0/0
    Neighbor priority is 1, State is FULL, 6 state changes
    Options is 0x12 in Hello (E-bit, L-bit)
    Options is 0x52 in DBD (E-bit, L-bit, O-bit)
    LLS Options is 0x1 (LR)
    Dead timer due in 00:00:38
    Neighbor is up for 00:04:58
    Index 1/1, retransmission queue length 0, number of retransmission 10
```

```
RP/0/RSP0/CPU0:XR3#show ospf 100 neighbor detail
! Output omitted for brevity
Neighbors for OSPF 100

 Neighbor 192.168.1.1, interface address 10.0.13.1
    In the area 0 via interface GigabitEthernet0/0/0/13
    Neighbor priority is 1, State is FULL, 6 state changes
    Options is 0x52
    LLS Options is 0x1 (LR)
    Dead timer due in 00:00:38
    Neighbor is up for 00:04:58
    Number of DBD retrans during last exchange 0
    Index 1/1, retransmission queue length 0, number of retransmission 16
    First 0(0)/0(0) Next 0(0)/0(0)
    LS Ack list: NSR-sync pending 0, high water mark 0
```

Example 21-5 demonstrates that a GR has just taken place on the neighboring router R1. Notice that R2 and XR3 do not terminate the adjacency session because of the previously negotiated GR agreement. During a GR event, IOS routers display the OOB resynchronization countdown timer for the recovery. If R1 does not respond by the end of the timer, R2 and XR3 will consider the connection down. IOS XR routers also use the OOB-Resync timer to track the state of the neighbor adjacency but the timer status is not included in the **show** command output.

Example 21-5 *OSPF Graceful Restart Initiated*

```
R2#show ip ospf neighbor detail
! Output omitted for brevity
 Neighbor 192.168.1.1, interface address 10.0.12.1
    In the area 0 via interface GigabitEthernet0/0
    Neighbor priority is 1, State is FULL, 12 state changes
    Options is 0x52 in DBD (E-bit, L-bit, O-bit)
    LLS Options is 0x1 (LR)
    oob-resync timeout in 00:00:39
    Dead timer due in 00:00:39
    Neighbor is up for 00:08:00
    Index 1/1, retransmission queue length 0, number of retransmission 98
    First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
```

```
RP/0/RSP0/CPU0:XR3#show ospf neighbor  detail
! Output omitted for brevity
Neighbors for OSPF 100

 Neighbor 192.168.1.1, interface address 10.0.13.1
    In the area 0 via interface GigabitEthernet0/0/0/13
    Neighbor priority is 1, State is FULL, 6 state changes
    LLS Options is 0x1 (LR)
    Dead timer due in 00:00:39
    Neighbor is up for 00:08:00
    Number of DBD retrans during last exchange 0
    Index 1/1, retransmission queue length 0, number of retransmission 0
    First 0(0)/0(0) Next 0(0)/0(0)
    LS Ack list: NSR-sync pending 0, high water mark 0
```

Example 21-6 demonstrates that a GR completed successfully 11 seconds ago. R1 recovered, and the neighbor adjacency was not reset on the R2 and XR3 side of the connection per the GR agreement.

Example 21-6 *OSPF Graceful Restart Completed*

```
R2#show ip ospf neighbor detail
! Output omitted for brevity
 Neighbor 192.168.1.1, interface address 10.0.12.1
    In the area 0 via interface GigabitEthernet0/0
    Neighbor priority is 1, State is FULL, 16 state changes
    Options is 0x52 in DBD (E-bit, L-bit, O-bit)
    LLS Options is 0x1 (LR), last OOB-Resync 00:00:11 ago
    Dead timer due in 00:00:34
    Neighbor is up for 00:08:37
    Index 2/2, retransmission queue length 0, number of retransmission 98\
RP/0/RSP0/CPU0:XR3#show ospf neighbor  detail
! Output omitted for brevity
Neighbors for OSPF 100

 Neighbor 192.168.1.1, interface address 10.0.13.1
    In the area 0 via interface GigabitEthernet0/0/0/13
    Neighbor priority is 1, State is FULL, 10 state changes
    LLS Options is 0x1 (LR), last OOB-Resync 00:00:11 ago
    Dead timer due in 00:00:34
    Neighbor is up for 00:08:37
    Number of DBD retrans during last exchange 0
    Index 2/2, retransmission queue length 0, number of retransmission 0
    LS Ack list: NSR-sync pending 0, high water mark 0
```

Note During the GR, the NSF-aware router does not clear the route table entries or the neighbor adjacency, and therefore the age of the routes predates the GR event as if nothing happened. The NSF-capable router performing the SSO restarts the routing process, so the neighbor adjacency and route table entries age is reset to zero on the restarting router.

IS-IS

There are two GR configuration modes available for IS-IS.

- **Cisco:** The Cisco IS-IS NSF mode does not perform any GR signaling, instead control plane checkpoint state information is propagated between the active and standby RPs. During a switchover, the restarting router reestablishes the IS-IS adjacency with the neighbor using the control plane state information from prior to the RP switchover. The neighboring router does not need to be NSF-aware as the entire process completes successfully as long as the IS-IS keepalive timers do not expire.
- **IETF:** RFC 3847 describes the IETF GR method. IS-IS signals a GR using a new Restart Option TLV 211 extension in the hello packet. When the NSF-capable router restarts, it signals a hello packet with a Restart Request (RR) bit set to 1. The NSF-aware router responds back with a hello packet with the Restart Acknowledgment (RA) bit set to 1. The Restart Option TLV is removed from the hello packets once the LSDB is synchronized between the two peers.

Example 21-7 demonstrates how to configure the IETF GR method for IS-IS.

Example 21-7 *IS-IS NSF Configuration*

```
IOS
router isis LAB
nsf ietf
```

```
IOS XR
router isis LAB
nsf ietf
```

The IOS command **show clns neighbor detail** and the IOS XR command **show isis neighbor detail** display whether the neighbor is NSF capable. Example 21-8 demonstrates that the neighboring router is capable of supporting an IS-IS GR.

Example 21-8 *Verifying IS-IS NSF Capabilities*

```
R2#show clns neighbors detail
System Id      Interface   SNPA                State  Holdtime  Type Protocol
R1             Gi0/0       000c.860a.9000      Up     24        L2   IS-IS
  Area Address(es): 49.0001
  IP Address(es):  10.0.12.1*
  Uptime: 00:25:23
  NSF capable
  Interface name: GigabitEthernet0/0
```

```
RP/0/RSP0/CPU0:XR3#show isis neighbors detail

IS-IS LAB neighbors:
System Id      Interface        SNPA           State Holdtime Type IETF-NSF
R1             Gi0/0/0/13       000c.860a.9000 Up    24       L2   Capable
  Area Address(es): 49.0001
  IPv4 Address(es): 10.0.13.1*
  Topologies: 'IPv4 Unicast'
  Uptime: 00:25:23
```

EIGRP

Enhanced Interior Gateway Routing Protocol (EIGRP) includes a Restart (RS) bit that allows for the signaling of a GR. When the RS bit is set to 1, the neighboring NSF-aware router knows that an RP switchover is about to take place on the NSF-capable router. Once the SSO event completes, the two routers synchronize route tables with the RS bit still enabled. The NSF-aware router sends an End-of-Table (EOT) signal to indicate that it has provided all the updates, at which point the two routers clear the RS bit from the EIGRP packets.

Example 21-9 demonstrates that the IOS command to enable GR on the NSF-capable router is **nsf**. On IOS XR routers, the NSF capability is enabled by default. The EIGRP mode command to disable NSF is **nsf disable**.

Example 21-9 *IS-IS NSF Configuration*

```
IOS (Classic AS Configuration)
router eigrp 100
nsf
```

```
IOS (Named Mode Configuration)
router eigrp LAB
 address-family ipv4 unicast autonomous-system 100
   nsf
```

Example 21-10 displays the EIGRP neighbor status on an NSF-aware router. The highlighted output indicates the neighbor has been up for 50 minutes but that an RP switchover occurred on R1 52 seconds ago.

Example 21-10 *Viewing EIGRP Neighbor Status*

```
R2#show ip eigrp neighbors detail
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface        Hold Uptime    SRTT   RTO  Q  Seq
                                             (sec)          (ms)       Cnt Num
0   10.0.12.1               Gi0/0              11 00:50:25    1   200  0  11
   Time since Restart 00:00:52
   Version 5.0/3.0, Retrans: 1, Retries: 0, Prefixes: 5
   Topology-ids from peer - 0
```

```
RP/0/RSP0/CPU0:XR3#show eigrp neighbors  detail
IPv4-EIGRP neighbors for AS(100) vrf default

H   Address                 Interface        Hold Uptime    SRTT   RTO  Q  Seq
                                             (sec)          (ms)       Cnt Num
0   10.0.13.1               Gi0/0/0/13         10 00:50:25    4   200  0  10
   Restart time 00:00:52
   Version 5.0/3.0, Retrans: 0, Retries: 0, Prefixes: 5
```

BGP

RFC 4724 describes BGP GR signaling. Enabling the features modifies the initial BGP open negotiation message to include GR capability code 64. The GR capability informs the neighboring router that it should not reset the BGP session and immediately purge the routes when it performs an SSO.

During an RP SSO event, the TCP connection used to form the BGP session is reset, but the routes in the RIB are not immediately purged. The BGP NSF-aware router detects that the BGP TCP socket has cleared, marks the old routes stale, and begins a countdown timer, while at the same time continuing to forward traffic using the route table information from prior to the reset. Once the NSF-capable router recovers, it forms a new TCP session and sends a new GR message notifying the NSF-capable router that it has restarted. The two routers exchange updates until the NSF-capable router signals the end-of-RIB (EOR) message. The NSF-aware router clears the stale countdown timer and any stale entries that are no longer present are removed.

Note An RP failure will cause the BGP session to temporarily reset on both sides of the connection. The session uptime will correlate to the RP failure. On the restarting router, the route table entries will be purged, but on the NSF-aware router, the learned routes will remain unchanged with an age that precedes the GR.

Unlike the other routing protocols, BGP routers are not NSF-aware or NSF-capable by default. The GR capability requires manual configuration on both sides of the session.

The steps for configuring BGP NSF are as follows:

Step 1. **Enable GR and NSF awareness.**

GR support is enabled for all peers with the IOS command **bgp graceful-restart** or the IOS XR command **graceful-restart** in BGP configuration mode. The IOS command to selectively enable GR per peer is **neighbor** *ip-address* **ha-mode graceful-restart**. In IOS XR, GR can be enabled or disabled per peer with the command **graceful-restart** [**disable**].

Step 2. **Set the restart time (optional).**

The GR restart time determines how long the router will wait for the restarting router to send an open message before declaring the neighbor down and resetting the session. The value may be changed with the IOS command **bgp graceful-restart restart-time** *seconds* or with the IOS XR command **graceful-restart restart-time** *seconds*.

The default value is 120 seconds.

Step 3. **Set stale route timeouts (optional).**

The stale routes timeout determines how long the router will wait for the end-of-record (EOR) message from the restarting neighbor before purging routes. The value may be changed with the IOS command **bgp graceful-restart stalepath-time** *seconds* or with the IOS XR command **graceful-restart stalepath-time** *seconds*. The default value is 360 seconds.

Note Enabling GR capabilities on an IOS router after the session has already been established will trigger the session to renegotiate and will result in an immediate session flap. In IOS XR, enabling GR is nondisruptive. The session will continue working with the previous setting until the session is manually reset and the capability is negotiated.

Example 21-11 demonstrates how to configure GR. R1 is using BGP per neighbor GR, and XR3 has the feature enabled globally for all sessions.

Example 21-11 *BGP Graceful Restart*

```
R1
router bgp 65001
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 neighbor 10.0.13.3 ha-mode graceful-restart
```

```
XR3
router bgp 65003
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
```

The command **show bgp ipv4 unicast neighbor** can be used to determine whether GR capabilities have been negotiated. Example 21-12 verifies that routers R1 and XR3 have successfully negotiated NSF capabilities with each other.

Example 21-12 *Verifying Negotiated GR and NSF Capabilities*

```
R1#show bgp ipv4 unicast neighbors 10.0.13.3 | i Graceful
    Graceful Restart Capability: advertised and received
  Graceful-Restart is-enabled, restart-time 120 seconds, stalepath-time 360 seconds
RP/0/RSP0/CPU0:XR3#show bgp ipv4 unicast neighbors 10.0.13.1 | i Graceful
  Graceful restart is-enabled
   Graceful Restart (GR Awareness): received
    Graceful Restart capability advertised and receive
```

Note It is common to have BGP and IGP routing protocols on the same routers. To avoid suboptimal routing during an RP failover, the protocols should having matching NSF capabilities configured.

Nonstop Routing

Nonstop routing (NSR) is an internal Cisco router feature that does not use a GR mechanism to signal to neighboring routers that an RP switchover has taken place. Instead, the primary RP is responsible for constantly transferring all relevant routing control plane information to the backup RP, including routing adjacency and TCP sockets. During a failure, the new RP uses the “checkpoint” state information to maintain the routing adjacencies and recalculate the route table without alerting the neighboring router that a switchover has occurred.

Figure 21-5 demonstrates an RP switchover on R1, which has SSO with NSR enabled. The routing protocol peering between R1 and R2 is unaffected by the RP failure. R2 is unaware that a failure has occurred on R1. Throughout the entire RP failover process, the traffic continues to flow between the two routers unimpeded using the Cisco Express Forwarding (CEF) forwarding table.

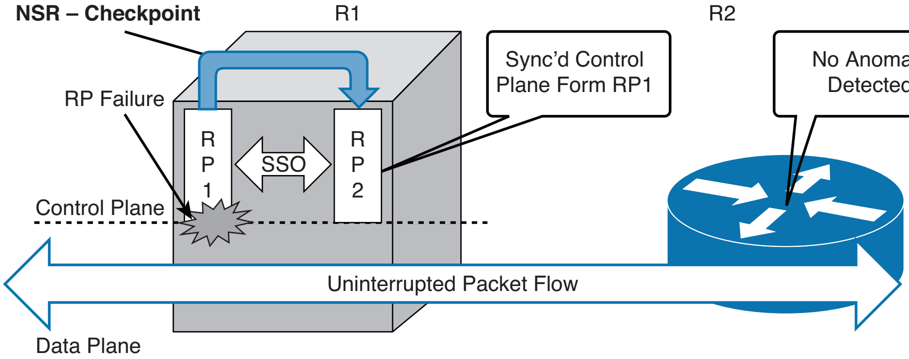

Figure 21-5 *SSO with NSR*

NSR's primary benefit over NSF is that it is a completely self-contained high availability solution. There is no disruption to the routing protocol interaction so the neighboring router does not need to be NSR- or NSF-aware. Table 21-2 outlines the expected routing protocol interaction between R1 and R2 when an RP switchover occurs on R1.

Table 21-2 *NSR Routing Protocol Interaction*

	SSO with NSR (R1 Internal RP1 --> RP2 Control Plane Checkpoint)			
	Neighbor Adjacency Reset		**RIB Recalculated Route Age Refreshed**	
	R1	**R2**	**R1**	**R2**
OSPF	No	No	Yes	No
IS-IS	No	No	Yes	No
EIGRP	NA			
BGP	No	No	Yes	No

NSR for OSPF

NSR for OSPF is enabled with the routing configuration mode command **nsr**. Example 21-13 shows how to enable NSR for OSPF.

Example 21-13 *OSPF NSR Configuration*

```
IOS
router ospf 100
 nsr
```

```
IOS XR
router ospf 100
 nsr
```

The IOS command to verify whether NSR is operational is **show ip ospf nsr**, and the IOS XR command is **show redundancy**. Example 21-14 demonstrates that the router is configured for NSR and that the backup RP is available for a switchover if required.

Example 21-14 *OSPF NSR Verification*

```
R1#show ip ospf nsr
 Active RP
 Operating in duplex mode
 Redundancy state: ACTIVE
 Peer redundancy state: STANDBY HOT
 Checkpoint peer ready
 Checkpoint messages-enabled
 ISSU negotiation complete
 ISSU versions compatible

 Routing Process "ospf 100" with ID 192.168.1.1
 NSR configured
 Checkpoint message sequence number: 2917
 Standby synchronization state: synchronized
! Output omitted for brevity
```

```
RP/0/RP0/CPU0:XR3#show redundancy summary
  Active/Primary   Standby/Backup
  --------------   --------------
   0/RP0/CPU0(A)    0/RP1/CPU0(S) (Node Ready, NSR: Ready)
   0/RP0/CPU0(P)    0/RP1/CPU0(B) (Proc Group Ready, NSR: Ready)
```

NSR for IS-IS

The IS-IS NSF operating mode **nsf cisco** is essentially NSR because the router uses SSO to checkpoint IS-IS control plane data between the RPs. The router does not signal a GR to its neighbor in this mode. Example 21-15 demonstrates how to enable NSF mode Cisco for IS-IS.

Example 21-15 *IS-IS NSF Mode Cisco Configuration*

```
IOS
router isis LAB
 nsf cisco
```

```
IOS XR
router isis LAB
 nsf cisco
```

NSR for BGP

IOS enables NSR for BGP on a per peer basis with the command **neighbor** *ip-address* **ha-mode sso**. The IOS XR BGP command **nsr** enables NSR for all neighbor sessions.

Note NSR and NSF/GR can be configured at the same time. Typically, NSR will take precedence over NSF. However, when deployed together with BGP on IOS, the router will attempt to use the NSF GR method over NSR. The IOS command **neighbor** *ip-address* **ha-mode graceful-restart disable** ensures that NSR is the active high availability feature used for the peering.

IOS XR routers give NSR preference over NSF GR when the two features are deployed in unison.

Example 21-16 demonstrates how to configure NSR for BGP. GR has been globally enabled within the BGP process. The GR capability has been disabled for the specific peer to ensure that SSO with NSR is employed.

Example 21-16 *BGP NSR*

```
R1
router bgp 65001
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 10.0.13.1 ha-mode sso
 neighbor 10.0.13.1 ha-mode graceful-restart disable
```

```
XR3
router bgp 65003
 nsr
```

The IOS command **show bgp ipv4 unicast sso summary** or the IOS XR command **show bgp ipv4 unicast sso summary** may be used to verify BGP NSR operational status, as demonstrated in Example 21-17.

Example 21-17 *BGP NSR Verification*

```
R1#show bgp ipv4 unicast sso summary
Load for five secs: 1%/0%; one minute: 1%; five minutes: 2%


   Total sessions with stateful switchover support-enabled: 1
   Total sessions configured with NSR mode and in established state: 1
! Output omitted for brevity
```

```
RP/0/RSP0/CPU0:XR3#show bgp ipv4 unicast sso summary
BGP router identifier 192.168.3.3, local AS number 65003
BGP generic scan interval 60 secs
Non-stop routing is-enabled
! Output omitted for brevity
Neighbor        Spk    AS   TblVer  SyncVer   AckVer NBRState     NSRState
10.0.13.1         0 65001      321        0      321 Established  None
Address Family IPv4 Unicast:
  Distance: external 20 internal 200 local 200
  Routing Information Sources:
    Neighbor          State/Last update received  NSR-State  GR-Enabled
    10.0.13.1          00:00:05                   NSR Ready  Yes
```

Nonstop Forwarding and Nonstop Routing Together

Routing protocols may use NSF and NSR together at the same time. NSR takes precedence for the IGP routing protocols, and NSF will be used as a fallback option where NSR recovery is not possible. For example, NSR does not support process restarts, and therefore there are benefits to deploying the two high availability features in tandem.

The IOS XR global command **nsr process-failures switchover** may be used when NSF is not enabled to force an RP failover when a routing process restarts.

Failure Avoidance

Routing protocols react to a link failure by recalculating a new best path to route around the problem. An unstable link can have a destabilizing effect on network convergence as the routers are constantly updating the routing table. To stabilize the overall topology, it may be desirable to temporarily remove the unstable prefix from the routing table as a method of failure avoidance until that section of the network stabilizes.

Route Flap Dampening

IP event dampening and BGP dampening are two features that suppress the effects of an oscillating or "flapping" link or route from routing protocol use. The two features follow the same route flap dampening (RFD) model. IP event dampening applies a penalty for a link transition, up to down. BGP dampening applies the penalty for unstable BGP routes. Each flap event incurs a penalty of 1000 until the dampening suppress threshold is met, at which point the routing protocols suppress advertising routes associated with the dampening. To prevent the routes from permanently being suppressed, an exponential decay mechanism reduces the associated dampening penalty. A half-life period determines how quickly the dampening penalty is reduced. Once the dampening penalty is below the reuse threshold, the route is unsuppressed and advertised throughout the network again.

Table 21-3 outlines the parameters used by route flap dampening.

Table 21-3 *RFD Parameters*

Parameter	**Description**
Half-life	The time it takes for a penalty to be decreased by half is the half-life period. The higher the half-life time interval, the longer dampening will penalize an unstable link. The exponential decay mechanism for reducing the penalty occurs every 5 seconds.
Suppress	A route is suppressed when its penalty exceeds the suppress value limit.
Reuse	The route is unsuppressed when the penalty for the flapping route falls below the reuse value threshold.
Restart penalty	The restart penalty is an optional IP event dampening parameter that assigns a penalty to an interface when it initializes for the first time after a router reload.
Max suppress time	This is the maximum amount of time a route prefix may be suppressed.
Max suppress penalty	The max suppress penalty is the largest penalty value a route may accrue. To ensure that a route is correctly suppressed, always use a suppress limit value that is less than the maximum penalty; otherwise, the route will not be susceptible to dampening. The value is not assigned in the command-line interface (CLI), but instead calculated by the router using the formula in Figure 21-6.

1 $\text{Max-penalty} = \text{reuse-limit} \times 2^{(\text{max-suppress-time/half-life})}$

2 $\text{Max-penalty} = 1000 \times 2^{(60/15)}$

3 $\text{Max-penalty} = 16000$

Figure 21-6 *Maximum Dampening Formula*

Figure 21-7 illustrates how IP event dampening applies a penalty to an interface that is unstable.

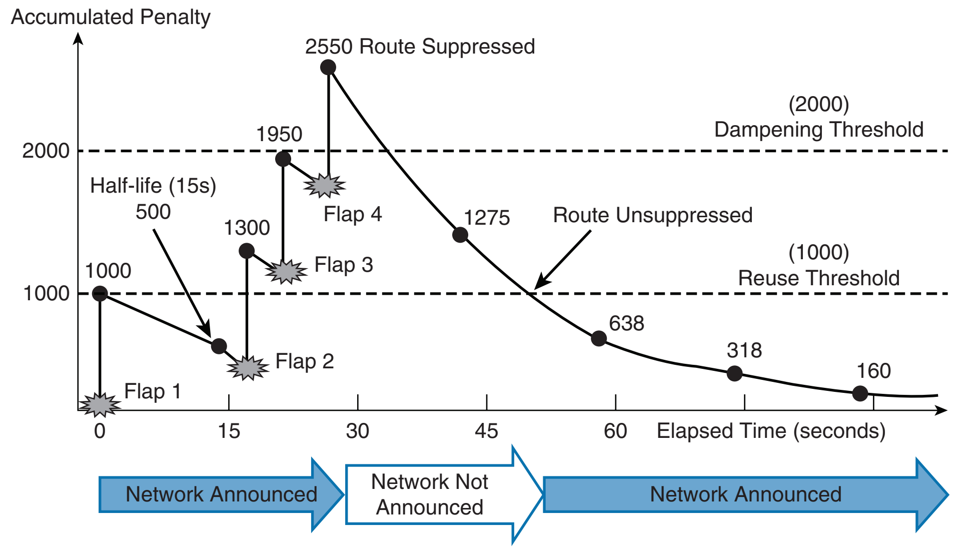

Figure 21-7 *Dampening*

The dampening in the illustration uses the following values:

- **Half-life:** 15 seconds
- **Reuse limit:** 1000
- **Suppress:** 2000
- **Max suppress time:** 60 seconds
- **Maximum penalty:** 16000

Initially, the interface has zero penalties accumulated. Link transitions increment a penalty of 1000. The dampening penalty is continuously decremented every 5 seconds using the exponential decay backoff algorithm. After 15 seconds (one half-life period), the value is reduced to 500. Over the next few seconds, the interface link-state transitions three more times. The fourth flap finally breaches the dampening suppression threshold of 2000, causing the routes associated with the interface to be suppressed. The routes are unsuppressed approximately 20 seconds later once the accumulated penalty value is reduced below the dampening reuse threshold.

IP Event Dampening

The IP event dampening feature dampens all routing protocol activity on an unstable interface. The dampened interface suppresses all routing activity, including advertising routes for the connected interface and forming neighbor adjacencies. In many designs, especially networks with link redundancy, it may be desirable to temporarily remove the unstable connection from the routing path as a method of failure avoidance until the link stabilizes. The feature is-enabled on a per interface basis with the IOS interface command **dampening** [*half-life-period reuse-threshold*] [*suppress-threshold max-suppress* [*restart-penalty*]] or with the IOS XR command **dampening** [*half-life-period* [*reuse-threshold suppress-threshold max-suppress*] [*restart-penalty*]].

Table 21-4 lists the default dampening values to be used when applying only the **dampening** command to the interface. Notice that the IP event dampening timer values are implemented in seconds for IOS and in minutes for IOS XR.

Table 21-4 *IP Event Damping Default Timers*

Parameter	**IOS**	**IOS XR**
Half-life	5 seconds	1 minute
Reuse	1000	750
Suppress	2000	2000
Max suppress time	Four times the half-life or 20 seconds	Four times the half-life
Restart penalty	0	0

Example 21-18 demonstrates how to configure IP event dampening. The IOS router is using the values from Figure 21-7, and the IOS XR router is using the default values from Table 21-4.

Example 21-18 *IP Event Dampening*

```
IOS
interface GigabitEthernet4/14
 dampening 15
```

```
XR
interface GigabitEthernet0/0/0/13
 dampening
```

The dampening status for an interface is viewed with the IOS command **show interface dampening** and the IOS XR command **show im dampening** *interface-type interface-number*.

Example 21-19 displays the status of the IOS and IOS XR interfaces after three successive interface flap events over a few seconds interval. Notice that the penalty exceeds the 2000 suppress threshold, so the interface manager process has notified the routing protocol to suppress the routes for the interface. The reuse timer indicates how long the interface needs to remain stable before the penalty value drops below the reuse threshold.

Example 21-19 *IP Event Dampening*

```
R1#show interfaces dampening
GigabitEthernet1/2/3
  Flaps Penalty    Supp ReuseTm   HalfL  ReuseV   SuppV  MaxSTm    MaxP Restart
      3    2678    TRUE      17      15    1000    2000      60   16000       0
```

```
RP/0/RSP0/CPU0:ASR9001-B#show im dampening interface gigabitEthernet 0/0/0/13
GigabitEthernet0/0/0/13 (0x04001380)
Dampening-enabled: Penalty 2678, SUPPRESSED (106 secs remaining)
  underlying-state:  Up
  half-life:         1        reuse:             750
  suppress:          2000     max-suppress-time: 4
  restart-penalty:   0
```

BGP Dampening

BGP route dampening applies a penalty for unstable route prefixes received from an eBGP neighbor. BGP dampening should be deployed with caution, as the feature may inadvertently filter business-critical networks. For example, dampening should not be applied to DNS subnets, especially root DNS servers.

BGP dampening may be applied for all routes or for a subset of routes using a route policy. The command to apply BGP dampening in IOS is **bgp dampening** [*half-life-time reuse suppress maximum-suppress-time* | **route-map** *route-map-name*] and IOS XR nodes use the command **bgp dampening** [*half-life* [*reuse suppress max-suppress-time*] | **route-policy** *route-policy-name*].

An IOS prefix-list or IOS XR prefix-set may be deployed with a BGP dampening route policy to specify which routes are eligible for dampening. Dampening values must be included within the route policy. The RFD eligible routes are identified using a permit statement in IOS or a pass statement in IOS XR, while the ineligible routes are denied in IOS or dropped in IOS XR.

Table 21-5 displays the default BGP dampening values for IOS and IOS XR

Table 21-5 *BGP Damping Default Timers*

Parameter	**IOS**	**IOS XR**
Half-life	15 minutes	15 minute
Reuse	750	750
Suppress	2000	2000
Max-suppress time	Four times the half-life or 60 minutes	Four times the half-life
Restart penalty	0	0

Note The RIPE Routing Working Group estimates that only 3 percent of all Internet prefixes are responsible for 36 percent of the BGP update messages on the Internet. To reduce routing churn, while at the same time penalizing unstable routes, the group recommends the minimum suppress limit threshold should be set to at least 6000 to 12,000 when using BGP dampening.

Example 21-20 demonstrates how to configure BGP dampening. The route policy BGP-DAMPENING is included in the configuration to demonstrate how to prevent prefixes of any length that reside in the 192.168.0.0/16 network from dampening. In the IOS example, notice that the prefix lists denies routes that are not subject to dampening. In the IOS XR RPL example, the dropped networks should not be dampened. All other network prefixes outside the range are passed and can be dampened by BGP.

The routers in the example are using the following dampening values:

- **Half-life time:** 15 minutes
- **Reuse value:** 750
- **Suppress value:** 6000
- **Max suppress time:** 60 minutes
- **Resulting max penalty:** 12,000

Example 21-20 *BGP Dampening Configuration*

```
R1
router bgp 65001
 neighbor 10.0.12.2 remote-as 65002
!
address-family ipv4
 bgp dampening route-map BGP-DAMPENING
 network 192.168.1.1 mask 255.255.255.255
 neighbor 10.0.12.2 activate
!
ip prefix-list DAMPENING seq 5 deny 192.168.0.0/16 le 32
ip prefix-list DAMPENING seq 10 permit 0.0.0.0/0 le 32
!
route-map BGP-DAMPENING permit 10
 match ip address prefix-list DAMPENING
 set dampening 15 750 6000 60
```

```
XR2
route-policy BGP-DAMPENING
  set dampening halflife 15 suppress 6000 reuse 750 max-suppress 60
  if destination in (192.168.0.0/16 le 32) then
    drop
  else
    pass
  endif
end-policy
!
router bgp 65002
 address-family ipv4 unicast
  bgp dampening route-policy BGP-DAMPENING
  network 192.168.3.3/32
 !
 neighbor 10.0.12.1
  remote-as 65001
  address-family ipv4 unicast
   route-policy pass in
   route-policy pass out
```

The command to view BGP dampening flap statistics in IOS is **show bgp ipv4 unicast dampening flap-statistics** and in IOS XR is **show bgp ipv4 unicast flap-statistics.**

Example 21-21 demonstrates how to view the BGP prefix flap statistics. In the example, the prefix 10.100.100.0/24 has flapped seven times over a span of less than 10 minutes, which has triggered BGP to dampen the prefix. The duration timer indicates when the first flap event occurred, and the reuse timer indicates when the prefix will be unsuppressed as long as the prefix remains stable.

Example 21-21   *BGP Dampen Flap Statistics*

```
R1#show bgp ipv4 unicast dampening flap-statistics
BGP table version is 99, local router ID is 192.168.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

    Network          From             Flaps Duration   Reuse    Path
*d  10.100.100.0/24  192.168.100.100   7    00:08:16   00:43:30 65005
```

```
RP/0/0/CPU0:XR2#show bgp ipv4 unicast flap-statistics
BGP router identifier 192.168.2.2, local AS number 65002
BGP generic scan interval 60 secs
BGP table state: Active
Table ID: 0xe0000000   RD version: 109
BGP main routing table version 109
Dampening-enabled
BGP scan interval 60 secs
```

```
Status codes: s suppressed, d damped, h history, * valid, > best
              i - internal, r RIB-failure, S stale, N Nexthop-discard
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network            From            Flaps   Duration  Reuse    Path
*d 10.100.100.0/24    192.168.100.100   7     00:08:16  00:43:30 65005 i
```

The command **show bgp ipv4 unicast dampened-paths** may be used to view actively dampened routes.

BGP dampened paths can be manually unsuppressed in IOS with the command **clear bgp ipv4 unicast dampening** *network mask* or in IOS XR with the command **clear bgp ipv4 unicast dampening** *network/length*.

Event-Driven Failure Detection

The first step in routing around a network failure is detecting that a problem has occurred. For example, a link failure can result in dropped packets for several seconds or minutes if the router continues to forward packets while waiting for the routing protocol hold timers to expire. Alternatively, event-driven detection, such as loss of signal (LoS), from a fiber cut, detects a link failure in fractions of a second and can significantly reduce the overall time required for routing convergence.

Carrier Delay

Carrier delay is the amount of time the router will wait before notifying upper layer protocols that the physical hardware interface has detected a LoS. The IOS interface command to tune the detection value is **carrier-delay** *seconds* and the IOS XR interface command is **carrier-delay up** *seconds* **down** *seconds*. The default timer value for IOS is 2 seconds, and for IOS XR it is 0 seconds.

Obviously, the lower the carrier detect value is set, the quicker the link failure will be detected and the sooner routing convergence may begin. Having the carrier delay value set to 0 turns off any delay, which might be detrimental to routing stability if the link is constantly flapping. Therefore, it is recommended that IP event dampening be used together with carrier delay to prevent constant routing recalculations. IOS XR and the latest versions of IOS XE also include a carrier delay up value. It is recommended that the up value is set to a higher value to ensure that the interface does not report active unless the link has been stable for a fixed amount of time. Example 21-22 demonstrates how to configure carrier delay.

Example 21-22 *Carrier Detect Configuration*

```
IOS
interface GigabitEthernet1/2/3
carrier-delay msec 0
```

```
IOS XR
interface GigabitEthernet0/0/0/13
carrier-delay up 3 down 0
```

Note The IOS optimization command **ip routing protocol purge interface** accelerates Forwarding Information Base (FIB) entry deletion by allowing routing protocols that are capable of responding to link failure to purge the route, instead of waiting for the less-efficient RIB process to walk the table to determine which routes are associated with the down interface. This command is enabled by default in Cisco IOS Software 15.1(2)S and later.

BFD

In some environments, no carrier detect signaling mechanism is available for quickly detecting whether the link between the two routers is down. Figure 21-8 illustrates three types of environments where a link failure may not occur on the directly connected interface. When the link failure is not directly connected, the router needs to rely on the routing protocol keepalive messages to determine remote-end neighbor reachability. This can take an unacceptably long amount of time by today's standards. For example, OSPF by default waits 40 seconds to declare a neighbor down.

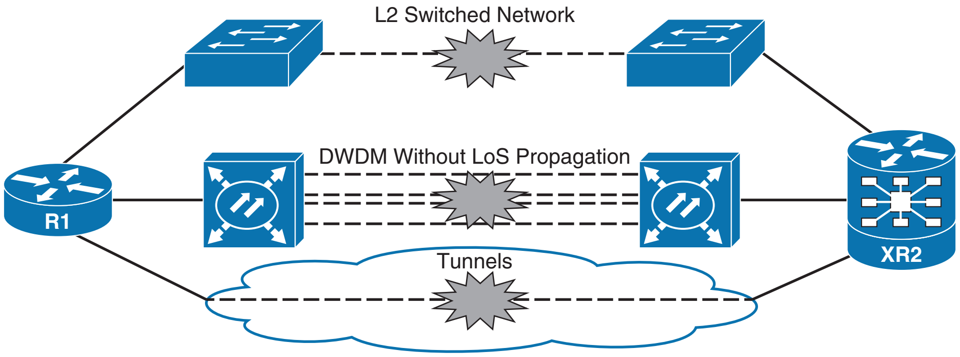

Figure 21-8 *Loss of Signal Detection Challenges*

One option for quickly identifying neighbor reachability loss is to set the hello and keepalive timers on the routing protocol to a very short interval. Introducing *fast hellos* does not always reduce the failure detection interval to a level where the network can route around the problem before time-sensitive applications notice the communication failure. In addition, fast hellos can tax the router's CPU and do not scale well as the number of neighbors sessions increase.

Bidirectional Forwarding Detection (BFD) is a simple lightweight hello protocol that routers can use on any media type for quickly detecting failures in the forwarding path. The BFD hello interval and hold timers for BFD are tunable, allowing for subsecond link failure detection.

Note BFD is commonly described as a lightweight hello protocol because of its small fixed packet header size. The packet's small size is less CPU-intensive to process than a more complex variable-length IGP routing protocol hello packet. On distributed routing platforms, the BFD packets are not punted to the RP CPU; instead, the line card processes the packets allowing for significant BFD session scalability.

BFD does not have an auto discovery mode. BFD treats routing protocols, such as OSPF, as clients for creating the BFD sessions. The routing protocol discovers the neighbor using its own detection mechanism and then uses this information to form the BFD session with the neighboring router. If a link failure is detected by BFD, the client routing protocol is notified. This allows OSPF to tear down the routing neighbor adjacency immediately, instead of waiting multiple seconds for the hold timers to expire.

Note Routing protocol hello and hold timers do not need to be modified to take advantage of BFD. In addition, if multiple routing protocols are enabled on the same interface with BFD, they will share the single BFD session.

RFCs 5880, 5881, 5882, and 5883 describe two modes of BFD operation:

- **Asynchronous mode:** In asynchronous mode, routers periodically send control packets to activate and maintain BFD sessions.

  Asynchronous mode is available in two submodes:

  - Asynchronous mode without echo
  - Asynchronous mode with echo

  Asynchronous mode with echo is the default operating mode for Cisco routers.

- **Demand mode:** In demand mode, the routers have an independent method of verifying connectivity to the other system. Once the BFD session is established, the routers are not required to share control packets with each other unless one side explicitly requests connectivity verification.

Asynchronous Mode Without Echo

BFD asynchronous mode without echo uses only control packets for negotiating session parameters and for detecting neighbor reachability. The BFD control packets have a UDP source port of 49152 and a destination port 3784.

Figure 21-9 illustrates an active BFD session between R1 and XR2. Notice that the BFD packet stream is unidirectional, meaning that there is not a request and response like with a client/server session. Instead, if the neighboring router does not receive the expected number of BFD packets in a row, the session is torn down.

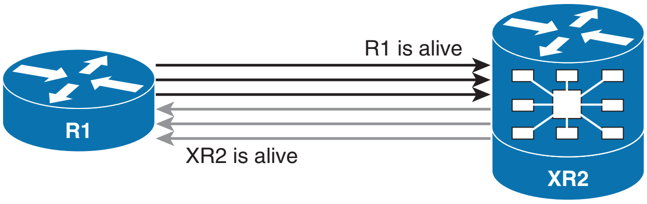

Figure 21-9 *BFD Asynchronous Mode Without Echo*

Asynchronous Mode with Echo

Asynchronous mode with echo uses both control UDP 3784 and echo UDP 3785 packets for forming and maintaining the BFD session. The control packets are sent at a default rate of 2 seconds (BFD slow-timers rate), while the echo packets are sent at a high rate for fast neighbor detection. The neighboring router receiving the BFD echo packets does not actually process the packets contents; instead, the packets are simply looped back to the originating router, unchanged. The primary benefit of the echo function is that it more accurately tests the forwarding path between the two routers in comparison with the BFD without echo.

Figure 21-10 illustrates an active BFD session between R1 and XR2. Notice that XR2 loops the echo packets back to the originating router, R1.

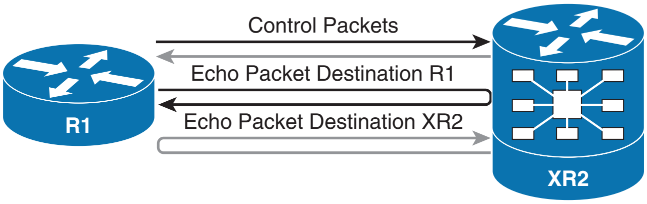

Figure 21-10 *BFD Asynchronous Echo Mode*

Note When unicast reverse path forwarding (uRPF) is enabled on an interface, asynchronous mode without echo is usually deployed to avoid inadvertently dropping looped echo packets. The IOS command to disable echo mode on an interface is **no bfd echo**. The IOS XR command is **bfd interface** *interface-type interface-number* **echo disable**.

Table 21-6 outlines the control and echo packets source and destination IP addresses and ports.

Table 21-6 *BFD IP Addresses and UDP Ports*

	Control Packets	**Echo Packets**
UDP destination	3784 (single) 4784 (multihop)	3785
UDP source	49152	3785
IP destination	Target IP address	IP address of outgoing interface
IP source	Local IP address identified by BFD client or IP address of outgoing interface	Selection order: **1.** Interface-specific configured address **2.** Global configured address **3.** Router ID **4.** IP address of outgoing interface

BFD Configuration for OSPF, IS-IS, and EIGRP

In IOS, three steps are required for enabling BFD support for a routing protocol.

Step 1. **Disable sending Internet Control Message Protocol (ICMP) redirect messages.**

BFD echo packets will trigger an IOS router to generate an ICMP redirect message back to the neighboring router because it believes a more optimal path exists out the local interface. The ICMP message is unnecessary and can trigger high CPU utilization. The interface command **no ip redirects** disables the sending of ICMP redirect messages.

Step 2. **Enable BFD on an interface.**

The BFD session parameters are enabled on the interface with the **bfd interval** *milliseconds* **min_rx** *milliseconds* **multiplier** *multiplier-value* command.

The **interval** *milliseconds* setting advertises the desired transmit interval for the BFD packets, while the **min-rx** *milliseconds* setting advertises the desired receive interval for BFD packets. The **multiplier** *multiplier-value* indicates how many missed BFD packets will trigger a session failure.

The packet interval range for IOS routers is 50 to 999 milliseconds. The negotiated timer values may be changed at any time and are negotiated in each direction. The router requesting the slowest rate determines the value.

Step 3. **Configure the routing protocol client.**

The routing protocol needs to be configured to use BFD. The routing protocol process configuration command **bfd all-interfaces** enables BFD on all interfaces associated with the routing protocol. BFD is enabled or disabled on a per-interface basis for EIGRP within the routing process:

- **Classic mode: bfd interface** *interface-type interface-number*

- **Named mode: af-interface** *interface-type interface-number* **bfd**

  BFD can be enabled or disabled on a per-interface basis for OSPF or IS-IS with the following interface parameter commands:

  ip ospf bfd [**disable**]

  isis bfd [**disable**]

In IOS XR, BFD activation and session parameters are both configurable under the dynamic routing protocol. The BFD session parameters are enabled within the routing protocol globally or on a per-interface basis. It is not necessary to disable **ipv4 redirects** because the feature is disabled by default on the interfaces. The command **bfd fast-detect** enables the routing protocol to react to BFD failure detection events. The commands **bfd minimum-interval** *milliseconds* and **bfd multiplier** *multiplier* manually set the desired transmit and receive packet interval. The range for IOS XR is 15 to 30,000 milliseconds.

Figure 21-11 illustrates the topology example for the BFD examples in this section.

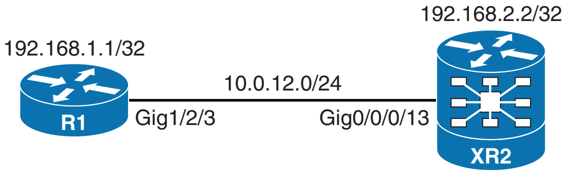

Figure 21-11 *BFD Topology*

Example 21-23 demonstrates how to configure the routers to form an OSPF routing adjacency using BFD. BFD is set to 150 ms for all interfaces. BFD packets are sent every 50 ms with three consecutive missed BFD echo packets triggering a session failure (150 ms).

Example 21-23 *OSPF BFD Configuration*

```
R1
interface GigabitEthernet1/2/3
 ip address 10.0.12.1 255.255.255.0
 no ip redirects
 bfd interval 50 min_rx 50 multiplier 3
!
router ospf 100
network 10.0.12.1 255.255.255.255 area 0
 bfd all-interfaces
```

```
XR2
router ospf 100
 bfd minimum-interval 50
 bfd fast-detect
 bfd multiplier 3
 area 0
  interface GigabitEthernet0/0/0/13
```

Note Less-sensitive BFD interval timers may be required for long-haul communication WAN links that have high latency. BFD timers of 500 ms * 3 or higher may be desirable for older SSO-capable router platforms that terminate the BFD session on the RP, instead of offloading the session to line card hardware. The higher timers are necessary to avoid a session flap during the time it takes the system control plane to failover to the secondary RP.

Example 21-24 demonstrates that the required OSPF interfaces have BFD enabled.

Example 21-24 *OSPF BFD Verification*

```
R1#show ip ospf interface GigabitEthernet 1/2/3 | i BFD
  Transmit Delay is 1 sec, State BDR, Priority 1, BFD-enabled
RP/0/RSP0/CPU0:XR2#show ospf interface gigabitEthernet 0/0/0/13 | i BFD
  BFD-enabled, BFD interval 150 msec, BFD multiplier 3
```

The IOS command **show bfd neighbors** [**details**] and the IOS XR command **show bfd session** [**details**] verify that the BFD session is active between the routers. Example 21-25 confirms that a BFD session successfully established between R1 and XR2.

Example 21-25 *BFD Session Status Verification*

```
R1#show bfd neighbors
IPv4 Sessions
NeighAddr                              LD/RD         RH/RS     State     Int
10.0.12.2                               1/2148073480 Up        Up        Gi1/2/3
```

```
RP/0/RSP0/CPU0:XR2#show bfd session
Interface           Dest Addr           Local det time(int*mult)      State
                                    Echo             Async
------------------- --------------- ---------------- ---------------- ----------
Gi0/0/0/13          10.0.12.1       150ms(50ms*3)    6s(2s*3)         UP
```

BFD Configuration for BGP

The BGP fast peering session deactivation feature (fallover) is paired with BFD detection to ensure that a session is deactivated as soon as the peer's IP address is no longer reachable.

The IOS command to enable BFD support for a specific neighbor session is **neighbor** *ip-address* **fall-over** [**bfd**].

Note BGP fast peering session deactivation does not require BFD and may be deployed for neighbor sessions that do not support BFD. Without BFD enabled, the session is torn down only if the local interface detects LoS or the route to the peer changes.

In IOS XR, BFD fast session deactivation is configured with the command **bfd fast-detect** under the neighbor peering. The packet timers may be configured once globally for all BGP sessions or individually for each peering.

Example 21-26 demonstrates how to configure BFD detection on the BGP peering between R1 and XR2.

Example 21-26 *BGP BFD Configuration*

```
IOS
interface GigabitEthernet1/2/3
 ip address 10.0.12.1 255.255.255.0
 no ip redirects
 bfd interval 50 min_rx 50 multiplier 3
!
router bgp 65001
 neighbor 10.0.12.2 remote-as 65002
 neighbor 10.0.12.2 fall-over bfd
!
 address-family ipv4
 neighbor 10.0.12.2 activate
```

```
XR
router bgp 65003
 neighbor 10.0.12.1
  remote-as 65001
  bfd fast-detect
  bfd multiplier 3
  bfd minimum-interval 50
  address-family ipv4 unicast
   route-policy PASS in
   route-policy PASS out
```

Example 21-27 demonstrates that the BGP routers are peering and that BFD fast detection is enabled.

Example 21-27 *OSPF BFD Configuration*

```
R1#show bgp ipv4 unicast neighbors 10.0.12.2 | i BFD
 BFD is configured. BFD peer is Up. Using BFD to detect fast fallover (single-hop).
RP/0/RSP0/CPU0:XR2#show bgp ipv4 unicast neighbors 10.0.12.1 | i BFD
  BFD-enabled (session up): mininterval: 50 multiplier: 3
```

Note Older router platforms do not support the usage of BFD and BGP NSF together. The latest IOS and IOS XE routers include support for RFC 5882 BFD control plane independent C bit, which allow for BFD and BGP NSF interoperability. The BGP command **neighbor** *ip-address* **fall-over bfd check-control plane-failure** allows the two features to work together at the same time. IOS XR routers support BGP NSF and BFD without the need for additional configuration requirements because BFD is offloaded to line card hardware making the sessions RP control plane independent.

BFD Configuration for BGP Multihop

BGP can use BFD multihop to help improve convergence when the peer is not directly connected. The BFD echo function is disabled when using BFD multihop per the guidelines of RFC 5883.

An IOS multihop session is defined using a BFD template and BFD map:

1. The template command **bfd-template multi-hop** *template-name* defines the multihop session parameters.
2. The BFD map command **bfd map ipv4** *destination-ip-prefix-length source-ip-prefix-length template-name* associates the source and destination address for the BFD session.

IOS XR requires identification of the specific line card that is to host the BFD multipath session with the command **bfd multipath include location** *node-id.*

Example 21-28 demonstrates how to configure an eBGP multihop session between R1's Loopback 0 interface (192.168.1.1/32) and XR2's Loopback 0 interface (192.168.2.2/32).

Example 21-28 *BGP Multihop BFD Configuration*

```
IOS
bfd map ipv4  192.168.2.2/32  192.168.1.1/32 BGP-XR2
bfd-template multi-hop BGP-XR2
 interval min-tx 50 min-rx 50 multiplier 3
!
router bgp 65001
 neighbor 192.168.2.2 remote-as 65002
 neighbor 192.168.2.2 ebgp-multihop 2
 neighbor 192.168.2.2 update-source Loopback0
 neighbor 192.168.2.2 fall-over bfd
 !
 address-family ipv4
 neighbor 192.168.2.2 activate
```

```
XR
bfd
 multipath include location 0/0/CPU0
```

```
!
router bgp 65002
 neighbor 192.168.1.1
  remote-as 65001
  bfd fast-detect
  bfd multiplier 3
  bfd minimum-interval 50
  ebgp-multihop 2
  update-source Loopback0
  address-family ipv4 unicast
   route-policy pass in
   route-policy pass out
```

Example 21-29 demonstrates an active BGP multihop session using BFD detection.

Example 21-29 *BGP Multihop BFD Verification*

```
R1#show bgp ipv4 unicast neighbors 192.168.2.2 | i BFD
 BFD is configured. BFD peer is Up. Using BFD to detect fast fallover (multi-hop).
```

BFD Configuration for Static Routes

Static routes may use BFD sessions for monitoring the neighbor router's next-hop reachability. Unlike the routing protocols, static route BFD does not have a dynamic method of learning the peer address for forming the BFD session. An IOS router can designate a peer for the BFD response with the interface command **bfd neighbor ipv4** *ip-address* or with a fully specified BFD-enabled static route.

The IOS command **ip route static bfd** *interface-type interface-number ip-address* identifies the neighbor router for the BFD session. Any additional static routes that include the BFD peer as a next-hop forwarding address will use BFD detection.

The IOS XR command to enable BFD for IPv4 static routes is **address-family ipv4 unicast** *ip-address next-hop* **bfd fast-detect** [**minimum interval** *interval*] [**multiplier** *multiplier*].

Example 21-30 demonstrates how to configure static routes to provide reachability between R1's and XR2's loopback interface. Notice that R1's configuration requires two static routes. The first static route identifies the target IP address 10.0.12.2 for forming the BFD session, and the second static route is for XR2's loopback address 192.168.2.2, which includes the next-hop forwarding address 10.0.12.2 that has BFD detection.

Example 21-30 *Static Route in Both Directions*

```
R1
interface GigabitEthernet1/2/3
 ip address 10.0.12.1 255.255.255.0
 no ip redirects
 bfd interval 50 min_rx 50 multiplier 3
 !
ip route static bfd GigabitEthernet1/2/3 10.0.12.2
ip route 192.168.2.2 255.255.255.255 GigabitEthernet1/2/3 10.0.12.2
```

XR2

```
router static
 address-family ipv4 unicast
  192.168.1.1/32 GigabitEthernet0/0/0/13 10.0.12.1 bfd fast-detect minimum-interval
  50 multiplier 3
```

The same BFD session verification commands apply to static routes. The IOS command **show ip static route bfd** and the IOS XR command **show bfd session destination** *ip-address* may be used to ensure that the static route's next-hop address is being monitored correctly by BFD.

Example 21-31 demonstrates how to verify the BFD session is active for the static route.

Example 21-31 *Static Route BFD Verification*

```
R1#show ip static route bfd
Codes in []: R - Reachable, U - Unreachable, L - Loop, D - Not Tracked

GigabitEthernet1/2/3 10.0.12.2 [R]
```

```
RP/0/RSP0/CPU0:XR2#show bfd session destination 10.0.12.1
Tue Apr 15 01:05:04.364 UTC
Interface           Dest Addr           Local det time(int*mult)      State
                                    Echo             Async
------------------- --------------- ---------------- ---------------- ----------
Gi0/0/0/13          10.0.12.1       150ms(50ms*3)    6s(2s*3)         UP
```

Example 21-32 demonstrates how to configure a default static route on the IOS XR router to use BFD, while the IOS router includes BFD neighbor awareness to allow the session to form.

Example 21-32 *BFD Session with XR2 Side Static Route*

R1

```
interface GigabitEthernet1/2/3
 no ip redirects
 bfd interval 50 min_rx 50 multiplier 3
 bfd neighbor ipv4 10.0.12.2
```

XR2

```
router static
address-family ipv4 unicast
  0.0.0.0/0 Gig0/0/0/13 10.0.12.1 bfd fast-detect minimum-interval 50 multiplier 3
```

Note The IOS interface configuration command **bfd neighbor** is not currently available on Cisco IOS XR Software.

Fast Routing Convergence

Once a router has detected a problem, it must still notify its neighbor of the failure, calculate a new best path, and then program the RIB/FIB forwarding tables before the network can converge. Historically, link-state routing protocol convergence timers focused on stability, instead of convergence speed. For example, before link-state advertisement (LSA) generation throttling was introduced, Cisco IOS Software waited a full 5 seconds after detecting a failure before generating an LSA message. For most modern applications, this is an unacceptably slow convergence time. The processing power today is significantly greater than when link-state routing protocols were first developed, and therefore the protocols can be safely modified to allow for faster convergence.

Figure 21-12 illustrates how the various routing protocol-tuning features contribute to event propagation and best path calculation after a link failure has been detected.

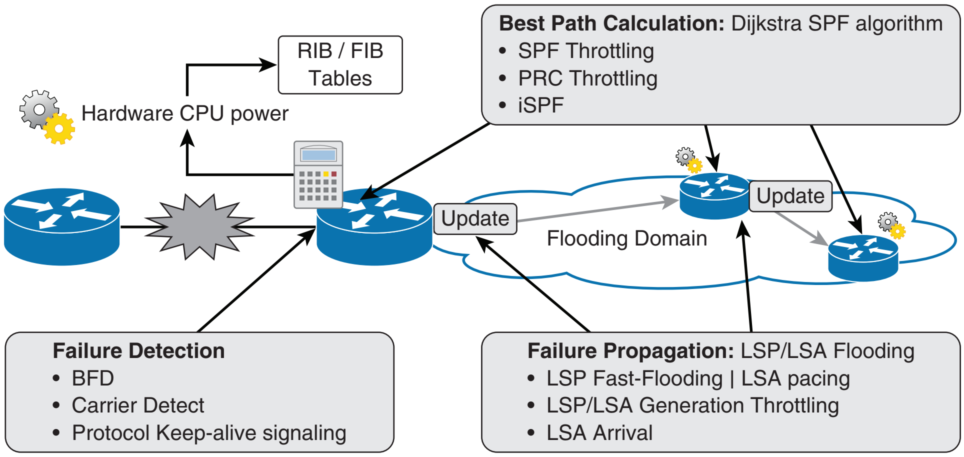
RIB / FIB Tables
Hardware CPU power
Best Path Calculation: Dijkstra SPF algorithm
• SPF Throttling
• PRC Throttling
• iSPF
Update
Update
Flooding Domain
Failure Detection
• BFD
• Carrier Detect
• Protocol Keep-alive signaling
Failure Propagation: LSP/LSA Flooding
• LSP Fast-Flooding | LSA pacing
• LSP/LSA Generation Throttling
• LSA Arrival

Figure 21-12 *Link-State Routing Protocol Convergence*

The OSPF and IS-IS link-state routing protocols propagate the failure information to neighboring routers in the same flooding domain to allow for the construction of a network connectivity map. Each router then independently performs a shortest path forwarding (SPF) calculation, called an *SPF run*, to determine the shortest path tree (SPT) for each destination network. The LSA/LSP flooding, along with the SPF run calculation, use a backoff throttling algorithm. The backoff timer allows for a quick reaction to a single event, but a delayed or more conservative stable reaction to a series of failures to reduce routing protocol churn. Throttling slows down convergence but is necessary for preventing uncontrolled routing fluctuations that can consume router resources and prevent routing convergence completion.

Table 21-7 describes the wait timers used by OSPF and IS-IS SPF backoff algorithm.

Table 21-7 *SPF Throttling Terminology*

OSPF	ISIS	Description
SPF start	SPF initial wait	The initial time delay after a topology change before performing an SPF calculation.
SPF hold	SPF second wait	The holdtime delay is the interval between two consecutive SPF calculations. The holdtime delay interval doubles after each SPF calculation and will continue doubling if a route flap event is detected during the timer countdown.
SPF max wait	SPF max wait	The maximum interval allowed between consecutive SPF calculations.

Figure 21-13 demonstrates how the exponential backoff algorithm works for SPF calculations. The illustration demonstrates SPF run throttling, but the concept is the same for link-state packet/advertisement (LSP/LSA) generation.

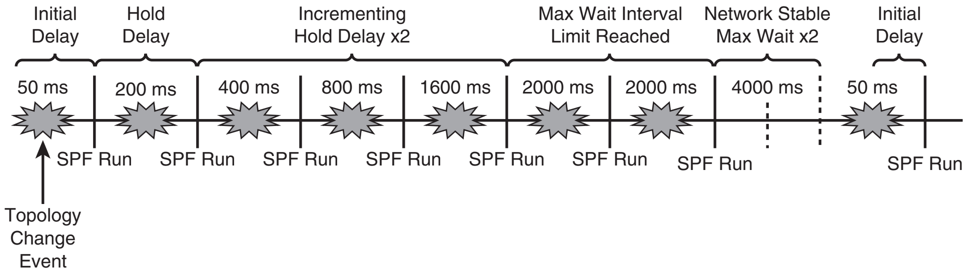

Figure 21-13 *SPF Exponential Backoff Throttle Mechanism*

The following IS-IS nondefault SPF throttling values are included for demonstrating the concept:

- **Initial holdtime start delay:** 50 milliseconds
- **Incrementing holdtime start delay:** 200 milliseconds
- **Maximum holdtime delay:** 2000 milliseconds

Notice how the router responds very quickly to the first route flap and performs an SPF calculation 50 milliseconds after learning of the event. The network is unstable, so each successive link flap delay doubles the incremental hold time delay of 200 milliseconds until finally the max holdtime delay limit of 2 seconds is reached. At this stage, the router waits a full 2 seconds before computing a best path for each successive link flap. The throttling hold timers do not reset until the network is stable for two full maximum wait holdtime intervals.

The next section of this chapter reviews the features that you can modify to reduce routing convergence times for the IS-IS and OSPF link-state protocols. It is important to note that IOS and IOS XR routers have different default LSP/LSA propagation and SPF processing values. Cisco IOS XR development engineers have tuned the values to allow convergence at a subsecond rate. Some of Cisco's largest service provider customers have safely used the values provided in the example configurations; but, of course, there is not a single "best practice" for all environments. The size of the network and the aggressiveness of the timer will determine the processing load that is placed on the router during periods of instability.

Modifying protocol timers can improve convergence times, but a high availability network requires a good network design. Where possible, the interfaces should be designated as network type point-to-point to avoid additional delays with the designated router (DR) and backup designated router (BDR) selection. A good design can improve convergence and stability by controlling LSA/LSP flooding through summarization and proper area placement.

IS-IS Convergence Tuning

Table 21-8 lists the default IS-IS LSP propagation timer values for IOS and IOS XR.

Table 21-8 *IS-IS Default LSP Flooding Values*

Parameter	IOS	IOS XR
Maximum LSP lifetime	7500 seconds (2 hours, 5 minutes)	1200 seconds (20 minutes)
LSP refresh interval	900 seconds (15 minutes)	900 seconds (15 minutes)
Fast flooding	disabled	10 LSPs per interface
LSP generation throttling	LSP initial wait: 50 ms LSP second wait: 5 seconds LSP max wait: 5 seconds	Initial wait: 50 ms Secondary wait: 200 ms Maximum wait: 5000 ms

The following configurable parameters influence IS-IS link-state PDU (LSP) flooding:

- **Maximum LSP lifetime:** The maximum amount of time a LSP can be in a router's database without being refreshed.

  IOS: **max-lsp-lifetime** *seconds*

  IOS XR: **max-lsp-lifetime** *seconds* [**level** {**1** | **2**}]

- **LSP refresh interval:** The interval at which the router will periodically refresh LSPs that it locally originates. Setting a high lifetime and refresh interval will reduce LSP generation, minimize routing protocol control traffic, and conserve router resources.

IOS: **lsp-refresh-interval** *seconds*

IOS XR: **max-lsp-lifetime** *seconds* [**level** {**1** | **2**}]

- **LSP fast flooding:** The fast flood feature tells the router how many LSP packets to flood to neighboring routers before starting the local SPF calculation. Fast flooding is recommended when using aggressive SPF throttle timers to ensure that the LSPs that triggered the topology change are flooded to neighbor routers. Flooding the packets before the local SPF computation is run will ensure that neighbor routers process the change simultaneously, enabling faster convergence time throughout the entire network.

  IOS: **fast-flood** [*lsp-number*]

  IOS XR: Automatically fast floods ten LSP by default. Individual interface fast flooding packet counts can be modified with the command **lsp fast-flood threshold** [*lsp-number*].

- **LSP generation throttling:** The amount of time the router waits before flooding new LSP updates to neighbors.

  IOS: **lsp-gen-interval** [**level-1** | **level-2**] *lsp-max-wait* [*lsp-initial-wait lsp-second-wait*]

  IOS XR: **lsp-gen-interval** [**initial-wait** *initial*] [**secondary-wait** *secondary*] [**maximum-wait** *maximum*] [**level** {**1** | **2**}]

Table 21-9 lists the default IS-IS SPF calculation interval values that affect best path calculation for IOS and IOS XR.

Table 21-9 *IS-IS Default SFP Calculation Intervals*

Parameter	IOS	IOS XR
SPF calculation throttling	SPF initial wait: 5500 ms SPF second wait: 5500 ms SPF max wait: 10 seconds	Initial wait: 50 ms Second wait: 200 ms Maximum wait: 5 seconds
PRC calculation throttling	PRC initial wait: 2000 ms PRC second wait: 5000 ms PRC max wait: 5 seconds	Initial wait: 50 ms Second wait: 200 ms Maximum wait: 5 seconds
Incremental SPF	Disabled	Disabled

The following parameters influence how long it takes IS-IS to process topology change notifications:

- **SPF calculation throttling:** SPF calculation throttling is the amount of time the router waits before performing a new best path calculation for the entire database. IS-IS performs a full SPF run only when the physical topology changes (for example, if a transit link between two routers fails within a level).

IOS: **spf-interval** [**level-1** | **level-2**] *spf-max-wait* [*spf-initial-wait spf-second-wait*]

IOS XR: **spf-interval** [**initial-wait** *initial* | **secondary-wait** *secondary* | **maximum-wait** *maximum*] [**level** {**1** | **2**}]

- **PRC calculation throttling:** A partial route calculation (PRC) calculates routes without needing to perform a full SPF computation. This may occur when the router learns new prefix information, yet the underlying physical network topology is the same. For example, a router changes the IP address of an interface.

  IOS: **prc-interval** *prc-max-wait* [*prc-initial-wait prc-second-wait*]

  IOS XR: PRC interval based on the settings of the **spf-interval**

Note Take special notice that the LSP generation, SPF, and PRC calculation throttling parameters entry order is not the same in IOS as it is in IOS XR. In addition, the maximum-wait hold time is set in seconds for IOS and in milliseconds for IOS XR. All other parameters are set in milliseconds.

- **Incremental SPF:** In large networks, the optional incremental SPF (iSPF) feature can accelerate routing table convergence. With iSPF enabled, the router does not perform a full SPT computation for the entire Level 1 (L1) or Level 2 (L2) database when a link failure is detected; instead, only the branches of the SPT that are affected by the topology change are recalculated.

  IOS: **ispf** {**level-1** | **level-2** | **level-1-2**}

  IOS XR: **ispf** [**level** {**1 | 2**}]

  Example 21-33 demonstrates how to configure the IS-IS tuning features discussed in this section. The example uses the following values to allow for subsecond routing convergence:

  - **Maximum LSP lifetime:** 65535 seconds
  - **LSP refresh interval:** 65000 seconds
  - **Fast flooding** Enabled, 10 packets
  - **LSP generation \ SPF \ PRC throttling:** 50 ms initial wait, 150 ms second wait, 5000 ms maximum wait

- **Incremental SPF:** Enabled for the Level 1 database.

Additional fast convergence features, such as BFD, carrier detect, and the IS-IS network type point-to-point are included in the sample configuration to demonstrate a full configuration. The IS-IS router is configured to form only L1 neighbor adjacencies in the example to avoid multiple adjacencies.

Example 21-33 *IS-IS Protocol Tuning Configuration*

IOS

```
interface GigabitEthernet1/2/3
 dampening 15
 ip address 10.0.12.1 255.255.255.0
 no ip redirects
 ip router isis LAB
 carrier-delay msec 0
 bfd interval 50 min_rx 50 multiplier 3
 isis network point-to-point
!
router isis LAB
 net 49.0001.0000.0000.0001.00
 is-type level-1
 ispf level-1
 fast-flood 10
 set-overload-bit on-startup 180
 max-lsp-lifetime 65535
 lsp-refresh-interval 65000
 spf-interval 5 50 150
 prc-interval 5 50 150
 lsp-gen-interval 5 50 150
 no hello padding
 nsf cisco
 bfd all-interfaces
```

IOS XR

```
interface GigabitEthernet0/0/0/13
 ipv4 address 10.0.12.2 255.255.255.0
 carrier-delay up 3 down 0
 dampening
!
router isis LAB
 set-overload-bit on-startup 180
 is-type level-1
 net 49.0001.0000.0000.0002.00
 nsf cisco
 lsp-gen-interval maximum-wait 5000 initial-wait 50 secondary-wait 150
 lsp-refresh-interval 65000
 max-lsp-lifetime 65535
 address-family ipv4 unicast
  ispf level 1
  spf-interval maximum-wait 5000 initial-wait 50 secondary-wait 150
 !
```

```
interface Loopback0
 passive
 address-family ipv4 unicast
 !
!
interface GigabitEthernet0/0/0/13
 bfd minimum-interval 50
 bfd multiplier 3
 bfd fast-detect ipv4
 point-to-point
 hello-padding sometimes
 address-family ipv4 unicast
```

OSPF Convergence Tuning

Table 21-10 lists the default OSPF LSA propagation timer values for IOS and IOS XR.

Table 21-10 *OSPF LSA Throttling*

Parameter	IOS	IOS XR
Maximum LSA lifetime (RFC 2328)	3600 seconds (1 hour)	3600 seconds (1 hour)
LSA refresh interval (RFC 2328)	1800 seconds (30 minutes)	1800 seconds (30 minutes)
LSA packet pacing	33 ms	33 ms
LSA generation throttling	Start interval: 0 ms Hold interval: 5000 ms Max interval: 5000 ms	Start interval: 50 ms Hold interval: 200 ms Max interval: 5000 ms
LSA arrival	1000 ms	100 ms

The following configurable parameters influence OSPF LSA flooding:

- **Flood reduction:** The OSPF maximum LSA lifetime (1 hour) and LSA refresh interval (30 minutes) are standards defined by the IETF in RFC 2328. To reduce excessive LSA flooding in large networks, flood reduction can be enabled per interface. Flood reduction disables LSA aging over the specified interface, reducing network overhead. Once enabled, LSAs will be flooded again only if there is a topology change.

  IOS: **ip ospf flood-reduction**

  IOS XR: **flood-reduction enable**

Note OSPF flood reduction requires that the adjacent router supports the DC bit (RFC 3883).

- **LSA packet pacing:** LSA packet pacing controls the interpacket timing gap between LSA updates. Pacing the packets ensures that an unstable network with a large amount of updates does not completely consume router resources.

  IOS: **timers pacing flood** *milliseconds*

  IOS XR: **timers pacing flood** *milliseconds*

- **LSA generation throttling:** The amount of time the router waits before flooding new LSA updates to neighbors. Throttling reduces routing protocol churn by reducing network updates during periods of network instability.

  IOS: **timers throttle lsa** *start-interval hold-interval max-interval*

  IOS XR: **timers throttle lsa all** *start-interval hold-interval max-interval*

- **LSA arrival:** The LSA minimal arrival timer determines the minimum amount of time that must pass before accepting the same LSA update.

  IOS: **timers lsa arrival** *milliseconds*

  IOS XR: **timers lsa min-arrival** *milliseconds*

Table 21-11 lists the default OSPF SPF calculation interval values that affect best path calculation for IOS and IOS XR.

Table 21-11 *OSPF Default SPF Calculation Intervals*

Parameter	IOS	IOS XR
SPF calculation throttling	No throttling	Start interval: 5000 ms Hold interval: 10000 ms Max interval: 10000 ms
PRC calculation throttling	No throttling	No throttling
Incremental SPF	Disabled	Disabled

The following parameters influence how long it takes OSPF to process topology change notifications:

- **SPF calculation throttling:** SPF calculation throttling is the amount of time the router waits after receiving a topology change before performing a new best path calculation for the entire network. Throttling is a protection mechanism that ensures that an unstable link does not result in uncontrolled routing churn. When building the SPT in OSPF, all intra-area routers and networks (LSA Type 1 and 2) are considered nodes on the SPT and interarea network summary routes, Autonomous System

Border Router (ASBR) summary, and autonomous system external redistributed prefixes (LSA Types 3, 4, 5) are considered the leaves, and therefore only LSA Type 1 or an LSA Type 2 will trigger a full SPF run.

IOS: **timers throttle spf** *spf-start spf-hold spf-max-wait*

IOS XR: **timers throttle spf** *start-interval hold-interval max-interval*

Note The IOS OSPF throttle timers are entered in a different order than the IS-IS command. In OSPF, the initial SPF run is entered first, whereas in IS-IS the value is entered second after the maximum wait interval.

- **PRC calculation:** OSPF supports partial route calculations for LSA Type 3, 4, and 5 topology changes. Cisco routers do not throttle PRC computations.
- **Incremental SPF:** Incremental SPF can reduce router CPU utilization by minimizing the scope of SPF calculations. A link-state change does not usually affect the SPT for every node in the network. With iSPF-enabled, a LSA Type 1 and 2 topology change does not trigger a full SPF run, only the branch of the tree that is affected is recalculated.

  IOS: **ispf**

  IOS XR: Not available

Note The iSPF feature only affects the SPT computation on the local router. On most modern routers, there is minimal benefit to enabling this feature because there is enough computational power to perform a full SPF run in the same amount of time as an iSPF.

Example 21-34 demonstrates how to configure OSPF protocol tuning to allow for subsecond convergence.

- **LSA packet pacing:** 10 packets
- **LSA generation \ SPF throttling:**
  - 50 ms initial wait
  - 150 ms incremental hold interval
  - 5000 ms max wait
- **LSA minimal arrival:**100 ms
- **Incremental SPF:** Enabled for IOS

Example 21-34 *OSPF Protocol Tuning Configuration*

IOS

```
interface GigabitEthernet1/2/3
 dampening 15
 ip address 10.0.12.1 255.255.255.0
 no ip redirects
 ip ospf flood-reduction
 ip ospf network point-to-point
 carrier-delay msec 0
 bfd interval 50 min_rx 50 multiplier 3
!
router ospf 100
 router-id 192.168.1.1
 ispf
 nsr
 nsf cisco
 timers throttle spf 50 150 5000
 timers throttle lsa 50 150 5000
 timers lsa arrival 100
 timers pacing flood 10
 passive-interface Loopback0
 network 10.0.12.1 0.0.0.0 area 0
 network 192.168.1.1 0.0.0.0 area 0
 bfd all-interfaces
```

IOS XR

```
interface GigabitEthernet0/0/0/13
 ipv4 address 10.0.12.2 255.255.255.0
 carrier-delay up 3 down 0
 dampening
!
router ospf 100
 nsr
 router-id 192.168.2.2
 nsf cisco
 timers throttle lsa all 50 150 5000
 timers throttle spf 50 150 5000
 timers lsa min-arrival 100
 timers pacing flood 10
 area 0
  interface Loopback0
   passive enable
  !
  interface GigabitEthernet0/0/0/13
```

```
bfd minimum-interval 50
bfd fast-detect
bfd multiplier 3
network point-to-point
flood-reduction enable
```

Cisco routers maintain a SPF run history, including the event that triggered the calculation. To view the OSPF SPF history, the IOS command is **show ip ospf statistics** [**detail**], and the IOS XR command is **show ospf statistics spf** [**detail**].

SPF Prefix Prioritization

The speed a router can program the RIB and FIB forwarding tables directly correlates to the processing power of its hardware. The prefix prioritization feature improves routing convergence for critical networks by allowing installation into the RIB based on priority. Prefix prioritization provides the greatest benefit in large networks. Instead of processing thousands of prefixes in a linear manner after completing the SPF run, the most important network prefixes are loaded into the route table and CEF forwarding table first.

Table 21-12 lists the default prefix priority for IOS and IOS XR.

Table 21-12 *Default Prefix Priority*

Protocol	IOS	IOS XR
IS-IS	**High:** None, may be matched by a tag **Medium:** /32 prefixes **Low:** All other prefixes	**Critical:** None, may be matched by an ACL **High:** None, may be matched by an ACL **Medium:** /32 prefixes **Low:** All other prefixes
OSPF	**High:** /32 prefixes **Low:** All other prefixes	**Critical:** None, may be matched by an RPL **High:** None, may be matched by an RPL **Medium:** /32 prefixes **Low:** All other prefixes

By default, IS-IS and OSPF give /32 prefixes special priority. The remaining prefixes default to the low-priority RIB processing queue. Customizing the priority queue disables the default priority for all prefixes. All prefixes that are not matched by the new policy, including /32 prefixes are moved to the low-priority queue.

Figure 21-14 illustrates a network using prefix prioritization to improve convergence on the IP TV subnet 10.0.1.0/24.

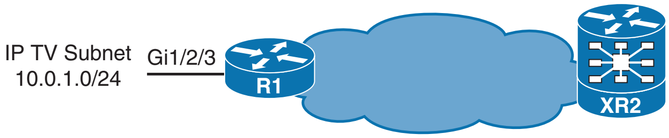

Figure 21-14 *Delay and Packet-Loss Sensitive IP TV Network*

In IOS, the IS-IS configuration mode command **ip route priority high tag** *tag-value* matches a prefix route tag value and assigns it to the high priority processing queue. Route tags may be assigned during redistribution or at the interface level with the command **isis tag** *tag-number*.

In IOS XR, prefixes are matched using an access list. The IS-IS address family command **spf prefix-priority** [**level** {**1** | **2**}] {**critical** | **high** | **medium**} {*acl-name* | **tag** *tag*} assigns the matched prefixes to the specified priority queue.

Example 21-35 demonstrates how to set the IP TV subnet to the high-priority queue in IOS and to the critical prefix priority queue in IOS XR. The /32 loopback prefixes are also manually set to the high-priority queue in both IOS and IOS XR. Notice that the IOS router performs matching using a route tag value, whereas in IOS XR, an ACL matches against the network address. To ensure that the IOS router correctly prioritizes the IOS XR loopback prefix, a tag is added on the IOS XR router's loopback interface.

Example 21-35 *IS-IS Prefix Prioritization*

```
IOS (R1)
interface GigabitEthernet1/2/3
 ip address 10.0.1.1 255.255.255.0
 ip router isis LAB
 isis tag 100
!
interface Loopback0
 ip address 192.168.1.1 255.255.255.255
 isis tag 100
!
router isis LAB
ip route priority high tag 100
```

```
IOS XR
ipv4 access-list HIGH
 10 permit ipv4 host 192.168.1.1 any
 20 permit ipv4 host 192.168.2.2 any
!
```

```
ipv4 access-list CRITICAL
 10 permit ipv4 10.0.1.0/24 any
!
router isis LAB
 is-type level-1
 net 49.0001.0000.0000.0002.00
 address-family ipv4 unicast
  spf prefix-priority critical CRITICAL
  spf prefix-priority high HIGH
 !
 interface Loopback0
  passive
  address-family ipv4 unicast
   tag 100
```

In IOS, prefix prioritization is enabled with the OSPF command **prefix-priority high route-map** *route-map-name*, and in IOS XR it is enabled with the command **spf prefix-priority route-policy** *route-policy-name.*

Note IOS sends high-priority OSPF prefixes to the RIB before low-priority prefixes of the same route type. For example, a high-priority external route will be processed before a low-priority external route, but after a low-priority interarea route.

IOS XR processes high-priority prefixes before low-priority prefixes regardless of route type.

Example 21-36 demonstrates how to set the IP TV subnet to the high priority in IOS and the critical priority in IOS XR. The policy also matches all /32 prefixes and sets them to the high-priority queue. Notice that OSPF is different from IS-IS because it uses a route map or RPL to match prefixes and set priority level.

Example 21-36 *OSPF Prefix Prioritization*

```
IOS
ip prefix-list PRIORITIZATION seq 5 permit 10.0.1.0/24
ip prefix-list PRIORITIZATION seq 10 permit 0.0.0.0/0 ge 32
!
route-map RIB-HIGH permit 10
 match ip address prefix-list PRIORITIZATION
!
Router ospf 100
prefix-priority high route-map RIB-HIGH
```

```
IOS XR
route-policy RIB-PRIORITY
  if destination in (10.0.1.0/24) then
    set spf-priority critical
  elseif destination in (0.0.0.0/0 ge 32) then
    set spf-priority high
  endif
end-policy
!
router ospf 100
spf prefix-priority route-policy RIB-PRIORITY
```

The IOS command to view the prefix priority level is **show ip ospf rib** [*ip-address* | *ip-address-mask*]. In IOS XR, the prefix priority level is viewable with the command **show route** [*ip-address/prefix-length*] **detail**.

Example 21-37 demonstrates how to view the priority level for the 10.0.1.0/24 prefix. In IOS, the priority has been set to high, and in IOS XR the priority has been set to critical for the prefix.

Example 21-37 *Verifying Prefix Priority*

```
R1#show ip ospf rib 10.0.1.0 | i Flags
     Flags: Connected, HiPrio
```

```
RP/0/0/CPU0:XR2#show ip route 10.0.1.0/24 detail | i RIB
  Route Priority: RIB_PRIORITY_NON_RECURSIVE_CRITICAL (4) SVD Type RIB_SVD_TYPE_LOCAL
```

BGP Convergence Tuning

This section reviews how to improve BGP convergence through next-hop tracking, accelerated advertisement updates, and tuning the underlying TCP protocol for faster update exchanges.

Next-Hop Tracking

When there is a network failure, it can take BGP a significant amount of time to detect that the next-hop forwarding address for a prefix is no longer valid. The BGP scanner process is responsible for verifying that a prefix's next-hop forwarding address information is still valid in the global route table. The scanner process runs once every 60 seconds. The scanner interval is configurable with the BGP configuration command **bgp scan-time** *seconds*; the increase in CPU load is significant, however, so modifying the scan time to a lower value is usually not recommended.

Note The process of examining every route's path information and attributes (NLRI) is commonly referred to as *walking the table*.

BGP next-hop tracking (NHT) is an event-driven failure detection mechanism where the RIB notifies BGP of the next-hop routing change, instead of having to wait for the BGP scanner process to periodically walk the table and discover the changes. BGP NHT is enabled by default in IOS and IOS XR. The BGP configuration command **bgp nexthop trigger enable** may be required in older versions of IOS software to turn on the feature.

BGP includes a trigger delay before reacting to a next-hop change to allow time for the IGP routing protocols to converge.

- **IOS:** 5 second delay
- **IOS XR:** 3 second delay (critical) / 10 second delay (non-critical)

IOS treats all routing changes for a BGP next-hop address the same, whereas IOS XR is more granular and considers a RIB route installation/withdrawal a critical event and a route metric change a noncritical event.

The NHT delay is configurable with the IOS BGP address family configuration command **bgp nexthop trigger delay** *seconds*. In IOS XR, the trigger delay is modified with the command **nexthop trigger-delay** {**critical** milliseconds | **non-critical** milliseconds}.

Note The next-hop trigger delay may be safely modified to a lower value of 0 or 1 if the IGP routing protocol has already been tuned for fast routing convergence. If the IGP has not been tuned or if there are frequent route flaps in the network, there could be a series of constant RIB changes. The reduced trigger delay can actually slow down convergence because the RIB implements NHT dampening. NHT dampening is not configurable and frequent next-hop route fluctuations can delay the BGP run by as much as 60 seconds.

The IOS global configuration command **ip routing protocol purge interface** is required if the NHT trigger delay is set to 0. This command ensures that if there is a local link failure that causes the next-hop address to be removed, the RIB process does not immediately notify BGP. Instead, the IGP protocols have a chance to converge and then notify BGP.

An optional route policy may be used for selective next-hop address tracking. The IOS command is **bgp nexthop route-map** *route-map-name* and the IOS XR address family command is **nexthop route-policy** *route-policy-name*. When creating an NHT route policy for IOS XR the policy must also include the source routing protocol of the next-hop prefix. The source routing protocol is conditionally matched in the route policy with the RPL configuration command **protocol in** (*protocol*). The route policy can be setup to allow only certain routes to be next-hop addresses for the BGP prefixes. For example, it may be desirable to only allow point-to-point (/30 or /31) and loopback interfaces (/32) as valid next-hop addresses. This type of policy intentionally avoids recursion to determine whether the next-hop address is reachable for the BGP prefix. A default or summary route will no longer be accepted as a valid option, potentially allowing for accelerated network convergence.

A thorough understanding of the BGP peering arrangement is necessary before deploying selective NHT. It is important that valid routes are allowed by the policy because if the route fails the next-hop policy evaluation, the BGP entry is marked as inaccessible, and the route is not installed in the global route table. Example 21-38 demonstrates a BGP path that is not usable because the next-hop address 10.0.23.2 fails the selective next-hop policy evaluation.

Example 21-38 *BGP Selective NHT Invalidating Next-Hop Address*

```
RP/0/0/CPU0:XR1#show bgp ipv4 unicast 192.168.1.1
! Output omitted for brevity
Paths: (1 available, no best path)
  Not advertised to any peer
  Path #1: Received by speaker 0
  Not advertised to any peer
  Local
    10.0.23.2 (inaccessible) from 0.0.0.0 (192.168.3.3)
      Origin incomplete, metric 3, localpref 100, weight 32768, valid, redistributed
      Received Path ID 0, Local Path ID 0, version 0
```

Example 21-39 demonstrates how to configure BGP NHT along with an optional selective next-hop tracking policy. The two routers are using a policy that limits candidate routes to prefixes with a /30 prefix length or greater and that are not already being learned from BGP. The trigger delay is set to 1 second. Notice that the IOS router sets the delay value using seconds and the IOS XR router uses milliseconds.

Example 21-39 *BGP with Selective Next-Hop Tracking*

```
IOS
ip prefix-list RECURSION-ROUTES seq 5 permit 0.0.0.0/0 le 29
!
route-map SELECTIVE-NHT deny 10
 match ip address prefix-list RECURSION-ROUTES
!
route-map SELECTIVE-NHT deny 20
 match source-protocol bgp 65000
!
route-map SELECTIVE-NHT permit 30
!
router bgp 65000
 address-family ipv4
  bgp nexthop route-map SELECTIVE-NHT
  bgp nexthop trigger delay 1
```

```
IOS XR
route-policy SELECTIVE-NHT
  if destination in (0.0.0.0/0 ge 30) and protocol in (connected, ospf 100, static)
then
    pass
  endif
end-policy
!
router bgp 65000
 address-family ipv4 unicast
  nexthop route-policy SELECTIVE-NHT
  nexthop trigger-delay critical 1000
  nexthop trigger-delay non-critical 1000
```

Minimum Route Advertisement Interval

A BGP router does not immediately propagate network updates to neighbors. Instead, a minimum route advertisement Interval (MRAI) is enforced that delays the advertisement of a prefix withdraw or announcement. RFC 4271 recommends a MRAI interval value of 30 seconds for eBGP neighbors and 5 seconds for iBGP sessions. These high values help reduce routing churn on the Internet and allow for fewer update messages to be sent but also slow down convergence. In IOS and IOS XR, the default MRAI values are 0 seconds for iBGP sessions and 30 seconds for eBGP.

Note In older versions of Cisco IOS Software, the MRAI values are 5 seconds for iBGP and 30 seconds for eBGP.

To accelerate BGP convergence the MRAI interval can be reduced or disabled all together when set to 0. The IOS command to change the interval is **neighbor** *ip-address* **advertisement-interval** *seconds*, and the IOS XR BGP neighbor command is **advertisement-interval** *seconds*.

TCP Performance

BGP uses the TCP protocol to form sessions and send updates. The larger the TCP packet payload, the more information that can be shared at one time and the sooner the BGP network will converge. To maximize the TCP payload size, the router can use maximum transmission unit (MTU) discovery so that the largest possible frame size can be identified for sending the packets. MTU discovery is enabled by default in IOS. The IOS command for BGP MTU path discovery is **bgp transport path-mtu-discovery**. MTU path discovery is not enabled by default in IOS XR and needs to be turned on with the global command **tcp path-mtu-discovery**.

To minimize TCP chattiness and achieve a higher BGP control session throughput the TCP window size can be increased to a larger value, such as 65535 bytes, with the IOS command **ip tcp-window-size 65535** and the IOS XR command **tcp window-size 65535**.

Note Multihop BGP sessions may encounter problems if a smaller MTU link is present in the path. The TCP MTU path discovery mechanism may not work when there are firewalls or asymmetric routing paths. The maximum segment size (MSS) can be set to a value within the tolerance of a transit router with the interface parameter command **ip tcp adjust-mss** *500-1460* on IOS nodes. IOS XR routers set the TCP MSS size for the entire router with the command **tcp mss** *68-10000*.

A TCP packet drop may slow down BGP convergence because the sending router has to retransmit all the packets from the previous window of data. Selective acknowledgments can be activated, allowing only the missing data segment to be resent. Selective acknowledgments are enabled in IOS with the global command **ip tcp selective-ack** and in IOS XR with the command **tcp selective-ack**.

Fast Reroute

Fast reroute (FRR) is a mechanism for reducing the router's failure reaction time by quickly redirecting traffic around a failure. The router accomplishes this task by precomputing a RIB and FIB repair path entry before the failure event. When a failure is detected, the traffic is forwarded to the repair path within tens of milliseconds without having to wait for the network to converge.

Loop-Free Alternate Fast Reroute

The purpose of FRR is to allow a router to quickly redirect traffic after detecting a link or adjacent node failure. The router, called the *protecting node*, accomplishes this task by precomputing a loop-free alternate (LFA) repair path prior to the primary path actually failing. Immediately following a primary path failure, traffic is rerouted over the repair path while the routing protocol converges around the failure and calculates a new best path.

LFA FRR does not perform signaling between neighboring routers and only provides local protection for traffic forwarding. The feature may be deployed on one router, a selection of routers, or on every router in the network.

Figure 21-15 illustrates a link failure between R1 and XR2. R1 has LFA FRR enabled and immediately redirects traffic to R3 without having to wait for the neighboring routers to recompute the network topology or even be aware that a failure has occurred. LFA FRR minimizes the local failure reaction time by eliminating the hundreds milliseconds of routing protocol delay that is normally present with IGP flooding and best path calculations. LFA FRR can redirect traffic to the precomputed backup path in less than 50 milliseconds once the failure event is detected.

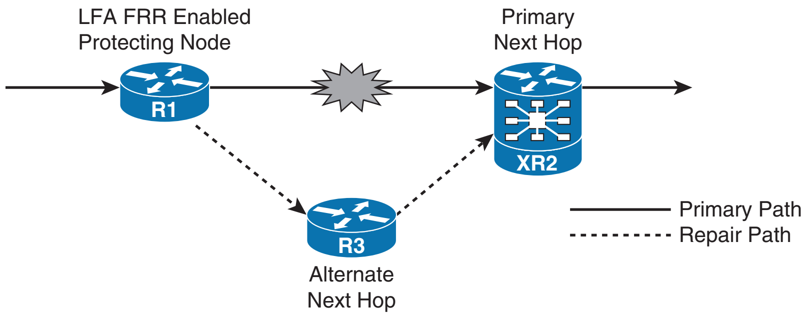

Figure 21-15 *IP Loop-Free Alternate Fast Reroute*

> **Note** Enabling LFA FRR does not impact the primary path convergence speed because the router always gives CPU priority to the primary path SPF calculation. The LFA FRR SPF run typically begins 500 ms after the network has converged.

Loop-Free Condition Rules

A network with multiple redundant links may not necessarily meet the criteria for building a precomputed repair path. RFC 5286 outlines a set of formulas for determining whether a path meets the loop-free condition required for a repair path.

The following terms are used to describe the inequality condition formulas:

N = Neighbor router

D = Destination

S = Source router performing calculation

E = Neighbor router being protected

PN = Pseudonode

Inequality 1: Loop Free

Inequality condition 1 is the minimum requirement for precomputing a repair path and meeting the LFA condition. The neighboring router (N) should not expect that the protecting node (S) has the better path to reach the destination prefix (D).

Formula: Distance (N, D) < Distance (N, S) + Distance (S, D)

Figure 21-16 demonstrates that the LFA repair path coverage depends on both the interface metric/cost and logical topology. The figure on the left passes the LFA available rule criteria, whereas the figure on the right does not because the metric between routers N, D is higher than the sum between routers N, S and S, D. The higher metric makes it impossible for S to use N as the repair path without temporarily introducing a short duration routing loop between the two routers while waiting for the routing protocol to converge using normal routing methods.

Note A short duration routing loop is commonly called a *microloop* or *µloop*. The duration of the loop usually depends the convergence time of the slowest router in the network.

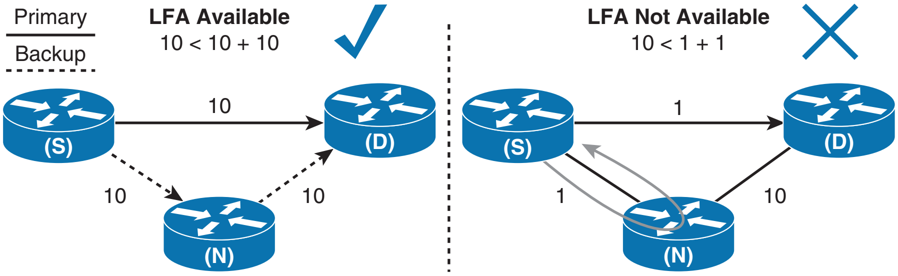

Figure 21-16 *Inequality Condition 1: Loop-Free Alternate Protection*

The repair path can be selected based on a set of more restrictive inequality conditions, which cover a wider range of failure scenarios. The more conditions that are met, the more resilient the repair path.

Inequality 2: Downstream Path

The neighboring router (N) is closer to the destination than the local router. This condition helps ensure that if there are multiple failures the neighbor router (N) will not form a loop and send traffic back to the computing router (S).

Formula: Distance (N, D) < Distance (S, D)

Figure 21-17 illustrates that the downstream path between N, D must be a smaller overall metric than the path between S, D.

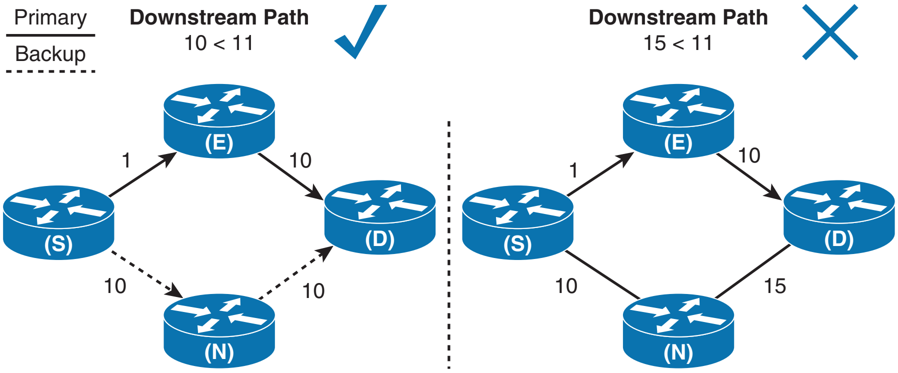

Figure 21-17 *Inequality Condition 2: Downstream Path Protection*

Inequality 3: Node-Protecting Loop-Free Alternate

The primary path between the source router (S) and the destination network (D) passes through the neighboring router (E). To ensure that a failure on E does not disrupt the repair path, the traffic directed to the next-hop protecting router (N) should not go through E.

Formula: Distance (N, D) < Distance (N, E) + Distance (E, D)

Figure 21-18 demonstrates that the repair path should not flow through router E to protect against a node failure on E.

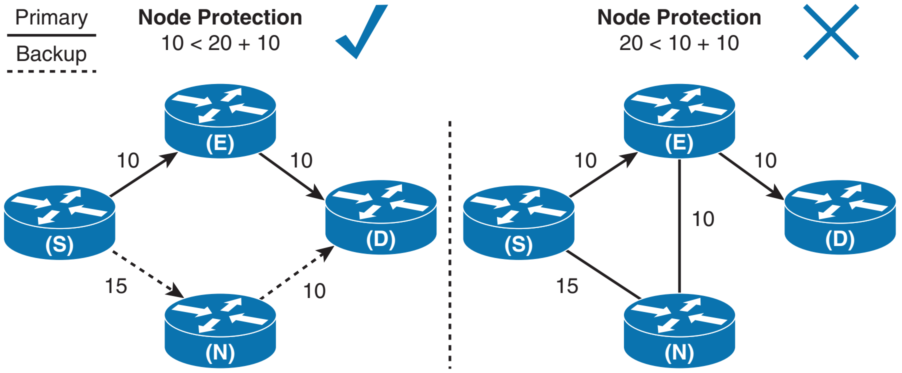

Figure 21-18 *Inequality Condition 3: Node Protection*

Inequality 4: Loop-Free Link-Protecting for Broadcast Links

A broadcast interface may have multiple attached routers to the same switch, and therefore it is possible to have different next-hop addresses for the primary path and repair path via the same link. The network failure may be the switch connecting the routers; therefore, the repair path should not cross the same broadcast network as the primary path.

Formula: Distance (N, D) < Distance (N, PN) + Distance (PN, D)

Figure 21-19 demonstrates that the repair path should not try to use the broadcast network to form a repair path. The pseudonode router (PN) represents the IS-IS DIS or OSPF DR for the LAN segment.

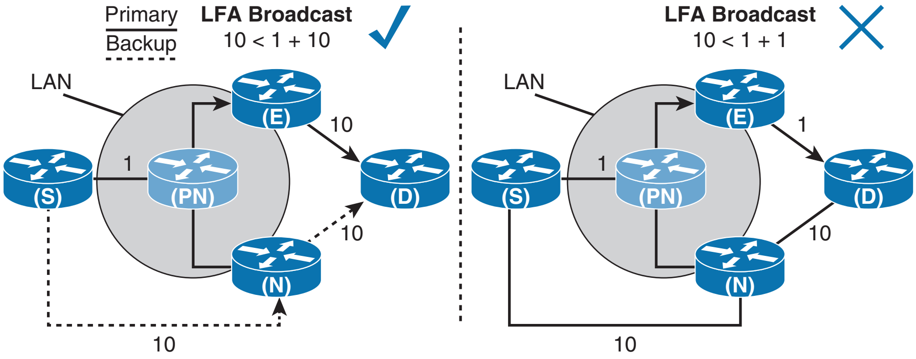

Figure 21-19 *Inequality Condition 4: LFA Broadcast Protection*

LFA Protection Modes

The two IP configuration modes available for creating the LFA repair path are per-link and per-prefix.

- **Per-link:** Per-link LFA protection examines the next-hop address of the primary link to determine whether a packet can be forwarded to another neighboring router without that neighbor in turn sending it back and forming a temporary routing loop while the IGP converges. Per-link LFA performs only a very simple inequality condition 1 "loop-free" check, and therefore there are minimal router CPU and memory requirements. All routes reachable through the primary protected link's next-hop address share the same repair path. Therefore, either all the prefixes using the net-hop address of the primary link are protected or none of the prefixes are protected. Spreading the load of multiple prefixes over diverse repair paths is not possible using per-link mode, and there is no guarantee that the repair path provides path or node protection for the destination network.
- **Per-prefix:** Per-prefix LFA performs a repair path computation for every prefix. This allows for an optimal repair path to be created for each destination network. Each prefix may have multiple candidate repair paths so a set of tiebreaker rules determine the best backup path. A tiebreaker attribute that eliminates all repair path candidates is skipped. Tiebreaker rule processing is sequential and finishes once a single path remains. If multiple candidate repair paths exist after processing the tiebreaker policy, the eligible repair paths are distributed among the prefixes to provide load sharing. Table 21-13 lists the repair path attributes available when using per-prefix LFA.

Table 21-13 *LFA Repair Path Attributes*

Tiebreak Attribute	Description
Shared risk link groups	Avoids candidate repair paths that belong to the same shared-risk link group (SRLG) as the primary path. An SRLG may be used to enhance repair path selection by identifying a set of links that share common resources and may fail simultaneously. For example, a set of interfaces may share the same fiber-optic path.
Equal-cost multi-path (ECMP) primary path	Avoids candidate repair paths that are not ECMPs. This option may be desirable when bandwidth overutilization over a single link is a concern.
ECMP secondary path	Avoids candidate repair paths that are ECMPs.
Interface disjoint	Avoid repair paths that share the same interface as the primary path.
Lowest repair path metric	Avoids candidate repair paths with a high metric. This option may not be desirable when high metric repair path provides more LFA protection coverage.
Line card disjoint	Avoids candidate repair paths that are on the same line cards as the primary path.
Node protecting	Avoids candidate repair paths that do not bypass the next-hop router of the primary path.
Broadcast interface disjoint	Avoids candidate repair paths that do not bypass the directly connected broadcast network of the primary path next hop.
Downstream	Avoids candidates repair paths that have downstream nodes with metrics to the destination higher than the protecting node metric to the destination.

RFC 6571 provides an excellent analysis of per-link and per-prefix LFA coverage for 11 of the most common service provider network topologies. The findings indicate that, on average, per-link LFA provided 67 percent repair path coverage, whereas per-prefix LFA provided 89 percent coverage. In general, per-prefix LFA provides the best overall LFA coverage and is the recommended mode of operation in pure IP environments.

Note Remote LFA (PQ) is a third method for calculating a repair path. Remote LFA is beyond the scope of this text because it not a pure native IP feature and requires Multiprotocol Label Switching (MPLS) to provide the LFA coverage.

Figure 21-20 is a reference network topology for the LFA FRR configuration examples.

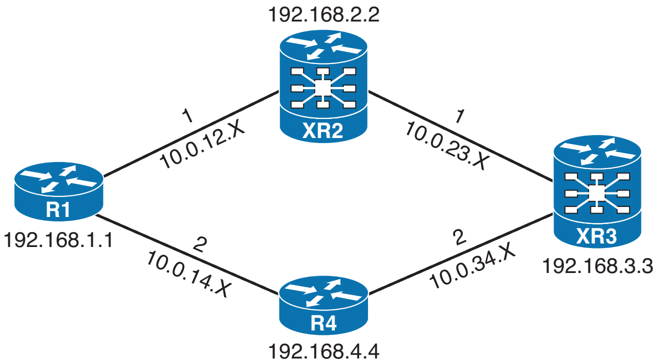

Figure 21-20 *LFA FRR Topology Example*

OSPF LFA FRR

The protecting OSPF router has the following restrictions for selecting a repair path:

- The candidate repair path's interface must participate in the same area as where the primary path's interface exists.
- The backup route must be of the same type, such as intra-area, interarea, external, or external NSSA, and the prefix must use the same metric type as the primary path.
- Repair paths over virtual links that terminate on the protecting router (headend) are not supported.

In IOS, there is only one required step for enabling LFA IPFRR for OSPF.

Step 1. **Enable loop-free alternate fast reroute.**

IOS routers support only per-prefix LFA. The OSPF command **fast-reroute per-prefix enable prefix-priority** [**area** *area-id*] *priority-level* enables per-prefix LFA. LFA may be enabled for the entire OSPF domain or for a specific area with the optional **area** keyword. The prefix priority level **low** enables repair path protection for all prefixes, and priority level **high** provides repair paths for /32 prefixes and or prefixes matched by an optional prefix-priority route map.

Step 2. **Modify repair path selection policy (optional).**

The OSPF command **fast-reroute per-prefix tie-break** *attribute* [**required**] **index** *index-level* configures the repair path selection policy. The attributes are evaluated in sequential order, starting with the lowest index number. The default IOS repair path policy selection order is displayed here. Modifying a tiebreak command removes the default rules, requiring a new manual configuration for all desired rules:

1. srlg index 10
2. primary-path index 20
3. interface-disjoint index 30

4. lowest-metric index 40
5. linecard-disjoint index 50
6. node-protecting index 60
7. broadcast-interface-disjoint index 70

Default repair path selection policy is viewable with the command **show ip ospf fast-reroute**.

Step 3. **Exclude an interface from the repair path (optional).**

The interface command **ip ospf fast-reroute per-prefix candidate disable** prevents an interface from being selected as next hop in a repair path.

Step 4. Disable protection on an interface (optional).

The interface command **ip ospf fast-reroute per-prefix protection disable** disables LFA next-hop protection for an interface. Primary routes that point to this interface will not be protected.

IOS XR supports per-prefix or per-link LFA:

Step 1. **Enable loop-free alternate fast reroute.**

LFA may be enabled in OSPF router, area, or interface configuration mode with the command **fast-reroute** {**per-link | per-prefix** } [**disable**]. The location at which the command is applied determines the scope of interfaces that are protected. Prefix protection can be selectively-enabled by priority levels with the command **fast-reroute per-prefix priority-limit** [**critical | high | medium**]. The default priority level for /32 prefixes is medium. All other prefixes default to low.

Step 2. **Modify repair path selection policy (optional).**

The OSPF command **fast-reroute per-prefix tiebreaker** *attribute* **index** *index-number* configures the repair path selection policy. The attributes are evaluated in sequential order starting with the lowest index number.

The default IOS XR repair path policy selection order is as follows:

1. Primary path index 10
2. Lowest metric index 20
3. Line card disjoint index 30
4. Node protection index 40

The command **show ospf** displays the default per-prefix tiebreaker rules.

Step 3. **Modify repair path interface candidate list (optional).**

The OSPF command **fast-reroute** {**per-prefix | per-link**} **use-candidate-only enable** excludes all interfaces from being candidate backup interfaces. Individual backup interfaces for the protected link are defined with the OSPF interface mode command **fast-reroute** {**per-prefix | per-link**} **lfa-candidate interface** *interface-type interface-number*.

Example 21-40 demonstrates how to figure OSPF per-prefix LFA FRR protection for routers R1 and XR3 in the topology example. All prefixes are eligible for protection in the example configuration.

Example 21-40 *OSPF LFA FRR Configuration*

```
IOS
router ospf 100
 fast-reroute per-prefix enable prefix-priority low
```

```
IOS XR
router ospf 100
 fast-reroute per-prefix
```

The IOS command to verify LFA prefix coverage is **show ip ospf fast-reroute prefix-summary**, and the IOS XR command is **show ospf statistics fast-reroute.**

Example 21-41 demonstrates how to view the LFA coverage for the example topology. Notice in the output that all of the networks are eligible for LFA protection, but only 66 percent have repair paths. The logic for 100 percent LFA coverage is described later in this section.

Example 21-41 *LFA Coverage for Figure 21-20*

```
R1#show ip ospf fast-reroute prefix-summary
! Output omitted for brevity
Interface        Protected    Primary paths    Protected paths Percent protected
                             All  High   Low   All  High   Low    All High  Low
Lo0                    Yes     0     0     0     0     0     0     0%   0%   0%
Gi2                    Yes     4     2     2     3     1     2    75%  50% 100%
Gi1                    Yes     2     1     1     1     0     1    50%   0% 100%

Area total:                    6     3     3     4     1     3    66%  33% 100%

Process total:                 6     3     3     4     1     3    66%  33% 100%
Process total:                 5     3     2     3     1     2    60%  33% 100%
Process total:                 6     3     3     4     1     3    66%  33% 100%
```

```
RP/0/0/CPU0:XR3#show ospf statistics fast-reroute
ospf_show_stats_ipfrr
OSPF 100 IPFRR Statistics:
 Number of paths:                            6
 Number of paths-enabled for protection :    6 (100%)
 Number of paths protected:                  4 (66%)
```

The IOS command to view the OSPF repair paths is **show ip ospf rib** or **show ip route repair-paths.** In IOS XR, the global route table includes (!) for the precomputed repair path. Example 21-42 provides a summary view of the OSPF repair paths.

Example 21-42 *Repair Path Summary*

```
R1#show ip ospf rib
            OSPF Router with ID (192.168.1.1) (Process ID 100)

                Base Topology (MTID 0)

OSPF local RIB
Codes: * - Best, > - Installed in global RIB
! Output omitted for brevity
*>  10.0.23.0/24, Intra, cost 2, area 0
      via 10.0.12.2, GigabitEthernet2
      repair path via 10.0.14.4, GigabitEthernet1, cost 5
*>  10.0.34.0/24, Intra, cost 4, area 0
      via 10.0.12.2, GigabitEthernet2
      repair path via 10.0.14.4, GigabitEthernet1, cost 4
      via 10.0.14.4, GigabitEthernet1
      repair path via 10.0.12.2, GigabitEthernet2, cost 4
```

```
RP/0/0/CPU0:XR3#show route
Codes: C - connected, S - static, R - RIP, B - BGP, (>) - Diversion path
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - ISIS, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, su - IS-IS summary null, * - candidate default
       U - per-user static route, o - ODR, L - local, G  - DAGR
       A - access/subscriber, a - Application route, (!) - FRR Backup path
! Output omitted for brevity
O    10.0.12.0/24 [110/0] via 10.0.34.4, 00:00:01, GigabitEthernet0/0/0/0 (!)
                  [110/2] via 10.0.23.2, 00:00:01, GigabitEthernet0/0/0/1
O    10.0.14.0/24 [110/4] via 10.0.34.4, 00:00:01, GigabitEthernet0/0/0/0
                  [110/4] via 10.0.23.2, 00:00:01, GigabitEthernet0/0/0/1
O    192.168.1.1/32 [110/0] via 10.0.34.4, 00:00:01, GigabitEthernet0/0/0/0 (!)
                    [110/3] via 10.0.23.2, 00:00:01, GigabitEthernet0/0/0/1
```

Example 21-43 demonstrates the forwarding entries for a single route. Notice that both the RIB and FIB are populated with the primary path (protected) and the repair path (backup).

Example 21-43 *Routing Table Entry For 192.168.3.3*

```
! RIB ENTRY
R1#show ip route 192.168.3.3
Routing entry for 192.168.3.3/32
  Known via "ospf 100", distance 110, metric 3, type intra area
  Last update from 10.0.12.2 on GigabitEthernet2, 11:43:39 ago
  Routing Descriptor Blocks:
  * 10.0.12.2, from 192.168.3.3, 11:43:39 ago, via GigabitEthernet2
      Route metric is 3, traffic share count is 1
      Repair Path: 10.0.14.4, via GigabitEthernet1

! FIB ENTRY
R1#show ip cef 192.168.3.3
192.168.3.3/32
  nexthop 10.0.12.2 GigabitEthernet2
    repair: attached-nexthop 10.0.14.4 GigabitEthernet1
```

```
! RIB ENTRY
RP/0/0/CPU0:XR3#show route 192.168.1.1
Routing entry for 192.168.1.1/32
  Known via "ospf 100", distance 110, metric 3, type intra area
  Routing Descriptor Blocks
    10.0.34.4, from 192.168.1.1, via GigabitEthernet0/0/0/0, Backup
      Route metric is 0
    10.0.23.2, from 192.168.1.1, via GigabitEthernet0/0/0/1, Protected
      Route metric is 3

! FIB ENTRY
RP/0/0/CPU0:XR3#show cef ipv4 192.168.1.1
192.168.1.1/32, version 56, internal 0x4000001 0x0 (ptr 0xacbe2e24) [1], 0x0
(0xacbde394), 0x0 (0x0)
local adjacency 10.0.23.2
 Prefix Len 32, traffic index 0, precedence n/a, priority 1
   via 10.0.34.4, GigabitEthernet0/0/0/0, 8 dependencies, weight 0, class 0, backup
[flags 0x300]
    path-idx 0 NHID 0x0 [0xacafc8c4 0x0]
    next hop 10.0.34.4
    local adjacency
   via 10.0.23.2, GigabitEthernet0/0/0/1, 8 dependencies, weight 0, class 0, pro-
tected [flags 0x400]
    path-idx 1 bkup-idx 0 NHID 0x0 [0xacf2a3b0 0x0]
    next hop 10.0.23.2
```

A review of the network topology reveals that the 192.168.x.x/32 networks advertised from directly adjacent neighbors fail the loop-free condition requirement rule 1.

Rule 1: Distance (N, D) < Distance (N, S) + Distance (S, D)

3 (R4, XR2) < 2 (R4, R1) + 1 (R1, XR2) – **Fails Condition**

3 (XR2, R4) < 1 (XR2, R1) + 2 (R1, R4) – **Fails Condition**

Figure 21-21 demonstrates that to provide 100 percent prefix repair path coverage, a link between XR2 and R4 is necessary.

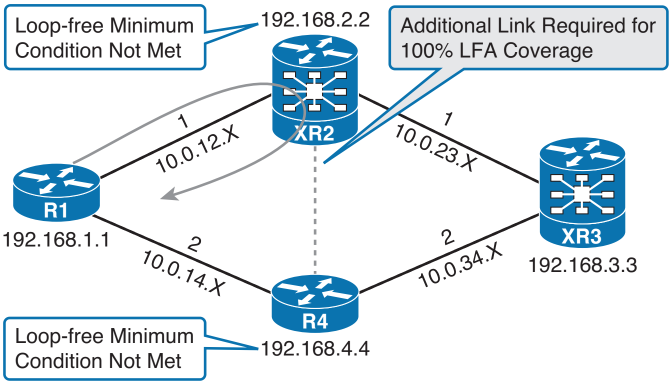
Loop-free Minimum Condition Not Met
192.168.2.2
Additional Link Required for 100% LFA Coverage
XR2
1
10.0.12.X
1
10.0.23.X
R1
192.168.1.1
XR3
192.168.3.3
2
10.0.14.X
2
10.0.34.X
R4
192.168.4.4
Loop-free Minimum Condition Not Met

Figure 21-21 *Requirements for 100 Percent LFA Coverage*

> **Note** Network modeling tools, such as Cisco Cariden, are helpful for identifying gaps in LFA repair path coverage.

IS-IS LFA FRR

Loop-free alternate FRR has the following requirements:

- Repair path links on the protecting node need to belong to the same level or area as the primary path interface.
- Cisco's IS-IS implementation of LFA protects only paths that go through point-to-point interfaces.

The IOS configuration procedure for IS-IS LFA FRR is as follows:

Step 1. **Enable loop-free alternate fast reroute.**

IOS routers only support per-prefix LFA. The IS-IS command **fast-reroute per-prefix** {**level-1 | level-2**} {**all | route-map** *route-map-name*} enables per-prefix LFA. All prefixes for a level may be protected using the **all** keyword or a subselection based on a route map match criteria. Both L1 and L2 routes can be protected by entering the command twice, once for each level.

Step 2. **Ensure protected interfaces are point-to-point.**

The interface command **isis network point-to-point** changes a broadcast interface to a point-to-point link.

Step 3. **Modify repair path selection policy (optional).**

The ISIS command **fast-reroute per-prefix tie-break** *attribute index-level* configures the repair path selection policy. The default IOS repair path policy selection order is as follows:

1. srlg-disjoint index 10
2. primary-path index 20
3. lowest-backup-path-metric index 30
4. linecard-disjoint index 40
5. node-protecting index 50

Default repair path selection policy is viewable within the configuration with the command **show run all | i tie-break**.

Step 4. **Exclude an interface from the repair path (optional).**

The interface command **isis fast-reroute candidate** {**level-1** | **level-2**} **disable** prevents an interface from being selected as next hop in a repair path.

On the primary protected interface, it is also possible to exclude a specific interface from being a candidate repair path using the command **isis fast-reroute exclude** {**level-1** | **level-2**} **interface** *interface-type interface-number.*

Step 5. **Disable protection on an interface (optional).**

The interface command **isis fast-reroute protection** {**level-1** | **level-2**} **disable** disables LFA next-hop protection for an interface. Primary routes that point to this interface will not be protected.

IOS XR supports per-prefix or per-link LFA for IS-IS:

Step 1. **Enable loop-free alternate fast reroute.**

LFA is-enabled in the IS-IS interface address family configuration mode with the command **fast-reroute** {**per-link** | **per-prefix**}. Similar to OSPF, IS-IS LFA prefix protection can be selectively enabled by priority level with the IS-IS address family command **fast-reroute per-prefix priority-limit** [**critical** | **high** | **medium**] {**level-1** | **level-2**}. The default priority level for /32 prefixes is medium. All other prefixes default to low.

Step 2. **Ensure that protected interfaces are point-to-point.**

The IS-IS interface mode command **point-to-point** changes a broadcast interface to a point-to-point link.

Step 3. **Modify repair path selection policy (optional).**

The IS-IS command **fast-reroute per-prefix tiebreaker** *attribute* **index** *index-number* **level** {**1** | **2**} configures the repair path selection policy.

The default IOS XR repair path policy selection order is as follows:

1. Primary path 10
2. Lowest metric 20
3. Line card disjoint 30
4. Node protection 40

Step 4. **Modify repair path interface candidate list (optional).**

The IS-IS address-family command **fast-reroute** {**per-prefix** | **per-link**} **use-candidate-only level** {**1** | **2**} excludes all interfaces from being candidate backup interfaces. Individual backup interfaces for the protected link are configurable within the IS-IS interface address family with the command **fast-reroute** {**per-prefix** | **per-link**} **lfa-candidate interface** *interface-type interface-number* **level** {**1** | **2**}.

Example 21-44 demonstrates how to configure IS-IS to use LFA FRR. In the configuration example, the routers are all in the same area and protect L1 routes. Notice that the interface network type is also set to point-to-point to fulfill the LFA requirements.

Example 21-44 *IS-IS LFA FRR Configuration*

IOS

```
interface GigabitEthernet1
 ip address 10.0.14.1 255.255.255.0
 no ip redirects
 ip router isis LAB
 isis network point-to-point
 isis metric 2
 bfd interval 50 min_rx 50 multiplier 3
!
interface GigabitEthernet2
 ip address 10.0.12.1 255.255.255.0
 no ip redirects
 ip router isis LAB
 isis network point-to-point
 isis metric 1
 bfd interval 50 min_rx 50 multiplier 3
!
router isis LAB
 net 49.0001.0000.0000.0001.00
 is-type level-1
 fast-reroute per-prefix level-1 all
 passive-interface Loopback0
 bfd all-interfaces
```

IOS XR

```
router isis LAB
 is-type level-1
 net 49.0001.0000.0000.0003.00
 address-family ipv4 unicast
 !
 interface Loopback0
  passive
  address-family ipv4 unicast
  !
 !
 interface GigabitEthernet0/0/0/0
  bfd minimum-interval 50
  bfd multiplier 3
  bfd fast-detect ipv4
  point-to-point
  address-family ipv4 unicast
   fast-reroute per-prefix
   metric 2
 !
 interface GigabitEthernet0/0/0/1
  bfd minimum-interval 50
  bfd multiplier 3
  bfd fast-detect ipv4
  point-to-point
  address-family ipv4 unicast
   fast-reroute per-prefix
   metric 1
```

The IOS and IOS XR command **show isis fast-reroute summary** provides an overview of the prefix LFA coverage.

The repair path information is viewable in the local RIB with the IOS command **show isis rib** [*ip-address* | *ip-address-mask*] or **show ip route repair-paths** [*ip-address* | *ip-address-mask*]. In IOS XR, the repair paths are viewable with the command **show route** or **show isis fast-reroute detail** [*prefix/prefix-length*]

Shared Risk Link Group

A shared-risk link group (SRLG) is a group of interfaces that share common physical paths or have a high likelihood of failing at the same time.

Figure 21-22 illustrates the physical connectivity and logical routed connectivity between R1 and XR2. Notice that the top two links share the same optical transport path. If the L1 network supporting these links has a failure then two out of the three L3 routing paths will also go down. To minimize packet loss, the LFA FRR routers should select the bottom path as the precomputed FRR repair path by assigning shared risk link group to the interfaces.

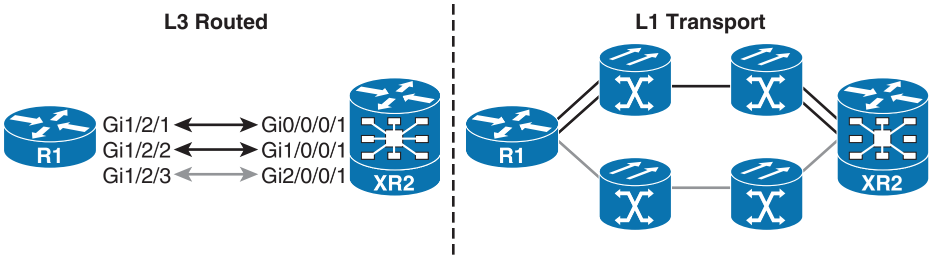

Figure 21-22 *L3 Routed Network and L1 Transport Network*

In IOS, an SRLG value is assigned with the interface command **srlg gid** *srlg-id*. In IOS XR, an SRLG value is assigned in SRLG configuration mode with the command **interface** *interface-type interface-number* **value** *value*. An interface may belong to zero, one, or multiple SRLGs.

Example 21-45 demonstrates how to configure SRLG value 12 for labeling the shared transport path between R1 and XR2. The tiebreak rules for OSPF and IS-IS LFA automatically prefer repair path candidates that are not members of the same SRLGs. LFA FRR only examines SRLG values that are locally configured on the protecting node and not the full path to the destination prefix.

Example 21-45 *Assigning SRLG to Interfaces*

```
IOS
interface GigabitEthernet1/2/1
 srlg gid 12
!
interface GigabitEthernet1/2/2
 srlg gid 12
```

```
IOS XR
srlg
 interface GigabitEthernet0/0/0/0
  value 12
 !
 interface GigabitEthernet0/0/0/1
  value 12
```

BGP Prefix-Independent Convergence

Prefix-independent convergence (PIC) improves convergence by installing a backup/alternate path in the RIB and CEF forwarding tables for BGP networks. The precomputed backup path allows for subsecond convergence by immediately redirecting traffic to the backup path upon detection of a primary path failure or withdrawal. Traffic remains on the alternate path until the network reconverges around the failure and a new best path is identified.

The BGP PIC fast reroute solution consists of two key modules, called *PIC core* and *PIC edge*:

- PIC core is a failure in the IGP path toward the BGP prefix next hop. PIC core uses a hierarchical FIB table and depends on the IGP for fast convergence.
- PIC edge is a fault at the edge of the network that may be caused by either a link or node failure, which results in the removal of the BGP prefix next-hop address. When this occurs, the PIC edge router redirects the traffic to the alternate BGP path.

Figure 21-23 illustrates the location for a PIC core or a PIC edge failure.

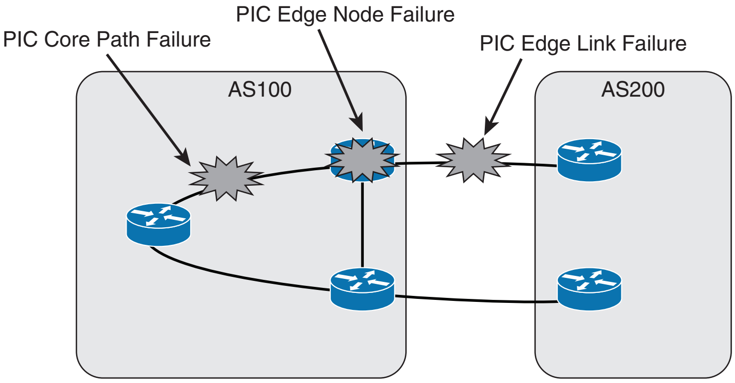

Figure 21-23 *BGP PIC Overview*

Note BGP PIC does not support SSO with NSF.

BGP PIC Core

BGP Internet tables can contain hundreds of thousands of prefix entries. The process of updating the FIB for such a large table can take many minutes to compile, as each individual prefix is evaluated and reprogrammed.

Figure 21-24 demonstrates a high-level view of a flat FIB table. Notice that each prefix has its own forwarding information directly associated to an outgoing interface (OIF) in a one-to-one correlation, which instructs the router how to forward the packet.

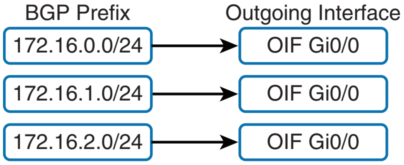

Figure 21-24 *Flat FIB*

BGP PIC core uses a hierarchical FIB that incorporates the concept of a shared path list and its in-place modification. In most environments, the majority of BGP prefixes point to a small number of IGP next-hop forwarding addresses. BGP PIC incorporates a pointer indirection between BGP and IGP entries so that a BGP prefix next hop that recurses to an IGP entry is immediately updated after the IGP converges.

Figure 21-25 demonstrates a hierarchical FIB shared path list table. Both single path and multipath BGP prefixes are represented to demonstrate how indirection and in place modification works. Only the objects of the FIB that are directly impacted by the topology change are updated. The shared path pointer enhancement in the hierarchical FIB allows convergence to remain constant regardless of the size of the BGP table, which is why the feature is aptly named prefix-independent convergence.

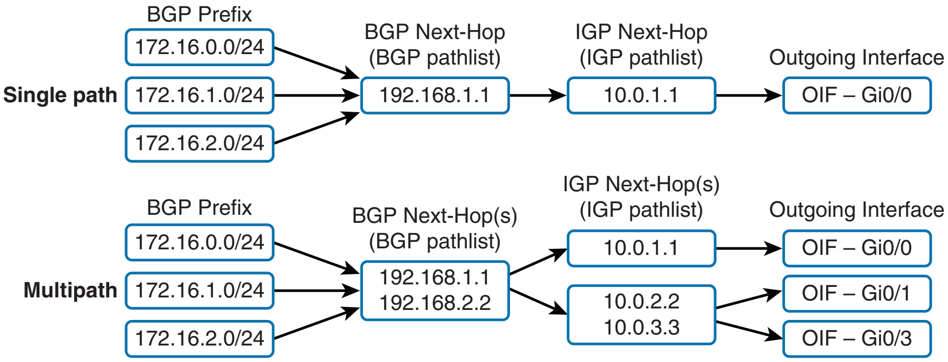

Figure 21-25 *Hierarchical FIB Shared Path List*

The IOS global configuration command **cef table output-chain build favor convergence-speed** enables BGP PIC core and in place FIB modifications on IOS platforms. IOS XR platforms do not require any additional configuration to enable PIC Core because they already use a hierarchical FIB by default.

Example 21-46 demonstrates how to configure BGP PIC Core on an IOS router.

Example 21-46 *BGP PIC Core*

```
IOS
cef table output-chain build favor convergence-speed
```

BGP PIC Edge

PIC edge redirects traffic to a backup path if the next-hop address of the primary path is lost due to a failure.

PIC edge is configured in IOS with the BGP address family command **bgp additional-paths install** and in IOS XR with the command **additional-paths selection route-policy** *route-policy-name*. Within the RPL, the **set** command **set path-selection backup 1 install** enables the installation of the backup path.

When the next-hop forwarding address fails, BGP normal routing behavior is to attempt to find the next longest matching path to reach the prefix. This behavior is not desirable when trying to achieve subsecond convergence and should be disabled when using PIC to ensure that traffic is immediately redirected to the backup path. The IOS BGP address family command to disable CEF recursion is **bgp recursion host**, and the IOS XR command is **nexthop resolution prefix-length minimum 32**.

Figure 21-26 provides a topology example that demonstrates the BGP PIC edge feature. The PIC configurations for R1 and XR2 are included in the example. BFD is not a strict configuration requirement for PIC edge, but is recommended for fast failure detection. For true bidirectional fast convergence, the routers in AS200 also need to have PIC enabled.

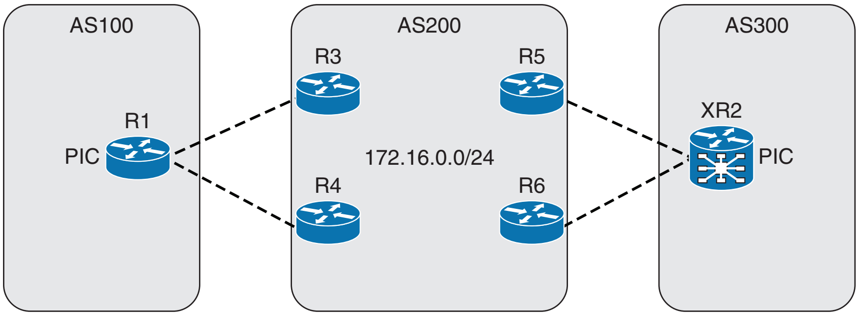

Figure 21-26 *PIC for Protecting Multihomed Network*

Example 21-47 demonstrates the configuration for BGP PIC edge.

Example 21-47 *BGP PIC Edge Configuration*

```
R1
router bgp 100
 neighbor 10.0.13.3 remote-as 200
 neighbor 10.0.13.3 fall-over bfd single-hop
 neighbor 10.0.14.4 remote-as 200
 neighbor 10.0.14.4 fall-over bfd single-hop
 !
 address-family ipv4
  bgp additional-paths install
  bgp recursion host
  network 192.168.1.1 mask 255.255.255.255
  neighbor 10.0.13.3 activate
  neighbor 10.0.14.4 activate
 exit-address-family
```

```
XR2
route-policy ALTERNATE
```

```
  set path-selection backup 1 install
end-policy
!
router bgp 300
 address-family ipv4 unicast
  additional-paths selection route-policy ALTERNATE
  nexthop resolution prefix-length minimum 32
  network 192.168.2.2/32
 !
 neighbor 10.0.25.5
  remote-as 200
  bfd fast-detect
  bfd multiplier 3
  bfd minimum-interval 50
  address-family ipv4 unicast
   route-policy PASS in
   route-policy PASS out
  !
 neighbor 10.0.26.6
  remote-as 200
  bfd fast-detect
  bfd multiplier 3
  bfd minimum-interval 50
  address-family ipv4 unicast
   route-policy PASS in
   route-policy PASS out
```

Similar to LFA, the BGP PIC backup path is viewable in the global route table using the IOS command **show ip route repair-paths** [*ip-address* | *subnet-mask*] or the with the IOS XR command **show route.**

Example 21-48 demonstrates that the alternate route is viewable in the BGP table with the command **show bgp ipv4 unicast** *ip-address*/*prefix-length*.

Example 21-48 *BGP Prefix Primary and Backup Path*

```
R1#show bgp ipv4 unicast 172.16.0.0/24
BGP routing table entry for 172.16.0.0/24, version 14
Paths: (2 available, best #2, table default)
  Additional-path-install
  Advertised to update-groups:
     1
  Refresh Epoch 1
  200
    10.0.14.4 from 10.0.14.4 (192.168.4.4)
```

```
      Origin IGP, localpref 100, valid, external, backup/repair , recursive-via-con-
nected
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1
  200
    10.0.13.3 from 10.0.13.3 (192.168.3.3)
      Origin IGP, localpref 100, valid, external, best , recursive-via-connected
      rx pathid: 0, tx pathid: 0x0
```

```
RP/0/0/CPU0:XR2#show bgp ipv4 unicast 172.16.0.0/24
BGP routing table entry for 172.16.0.0/24
Paths: (2 available, best #1)
  Advertised to update-groups (with more than one peer):
    0.2
  Path #1: Received by speaker 0
  Advertised to update-groups (with more than one peer):
    0.2
  200
    10.0.25.5 from 10.0.25.5 (192.168.5.5)
      Origin IGP, localpref 100, valid, external, best, group-best, import-candidate
      Received Path ID 0, Local Path ID 1, version 7
      Origin-AS validity: not-found
  Path #2: Received by speaker 0
  Not advertised to any peer
  200
    10.0.26.6 from 10.0.26.6 (192.168.6.6)
      Origin IGP, metric 0, localpref 100, valid, external, backup, add-path,
import-candidate
      Received Path ID 0, Local Path ID 2, version 12
      Origin-AS validity: not-found
```

Example 21-49 demonstrates that the IOS BGP table includes an additional *b* backup path status code for the alternate path for the prefix.

Example 21-49 *IOS BGP Table*

```
R1#show bgp ipv4 unicast
BGP table version is 13, local router ID is 192.168.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *b  172.16.0.0/24    10.0.14.4                              0 200 i
```

```
*>                   10.0.13.3                              0 200 i
*>  192.168.1.1/32   0.0.0.0                  0         32768 i
*b  192.168.2.2/32   10.0.14.4                              0 200 300 i
*>                   10.0.13.3                              0 200 300 i
```

BGP PIC Edge Link and Node Protection

The level of protection provided by BGP PIC edge depends on the location of the failure. If the failure is a link fault, the only routers that require backup paths are the autonomous system edge routers. However, if the edge router fails, the core routers will require a prepopulated alternate path in order to direct traffic around the failure.

Figure 21-27 illustrates link and node protection for BGP PIC edge. For PIC to succeed when there is an edge failure, the directly connected device requires a prepopulated alternate path to route around the failure.

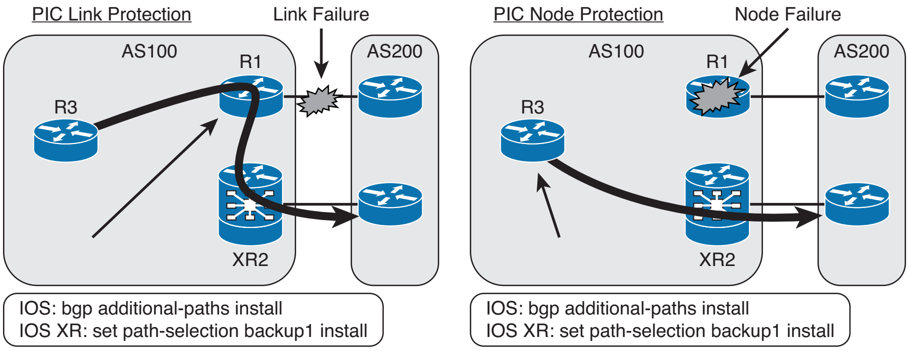
PIC Link Protection
Link Failure
AS100
AS200
R1
R3
XR2
IOS: bgp additional-paths install
IOS XR: set path-selection backup1 install
PIC Node Protection
Node Failure
AS100
AS200
R1
R3
XR2
IOS: bgp additional-paths install
IOS XR: set path-selection backup1 install

Figure 21-27 *BGP PIC Link and Node Protection*

BGP PIC Edge with Next-Hop-Self

BGP edge routers that use next-hop-self to advertise prefixes into the network may inadvertently hide link failures and prevent fast reroute from succeeding.

Figure 21-28 illustrates how a routing loop may form when using next-hop-self to advertise prefixes. The internal network is blind to the reachability state of the link connected to R1. When this link fails, the forwarding address of the BGP prefix from AS200 remains unchanged because the next-hop-self feature is using the loopback address of R1. R1's alternate path is via XR2. The internal routers continue forwarding traffic to R1 after the failure event. Because R1 does not have a direct link to XR2, traffic is sent back to the core network forming a continuous loop until routing convergence completes using normal BGP mechanisms.

The solution to this problem is to have a direct link or to tunnel the traffic directly to XR2 so that it may forward the traffic off-network without encountering the routing loop of the internal network.

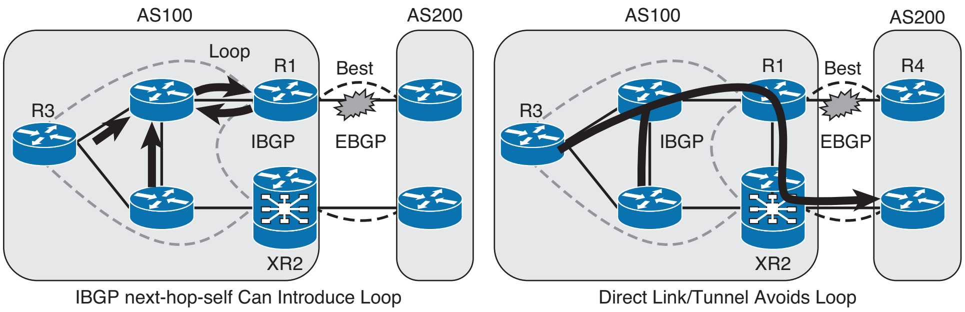
AS100
AS200
Loop
R1
Best
R3
IBGP
EBGP
XR2
IBGP next-hop-self Can Introduce Loop
AS100
AS200
R1
Best
R4
R3
IBGP
EBGP
XR2
Direct Link/Tunnel Avoids Loop

Figure 21-28 *BGP PIC with Next-Hop-Self*

BGP Advertise Best External

The default behavior of BGP is to advertise only the single best path for a BGP prefix. Unfortunately, a router that has knowledge of only one path cannot participate in PIC because there is not an alternate path to converge to.

Figure 21-29 demonstrates a problem when using local preference to set a primary and backup path to reach AS200. XR2 has applied a route policy that sets the local preference for the 172.16.0.0/24 network to 5000. R1 observes that XR2 has the most desirable path to reach 172.16.0.0/24, so it withdraws its path to reach the network. Now the only known path to reach 172.16.0.0/24 on R3 and XR2 is via XR2, preventing the installation of a backup path for fast convergence.

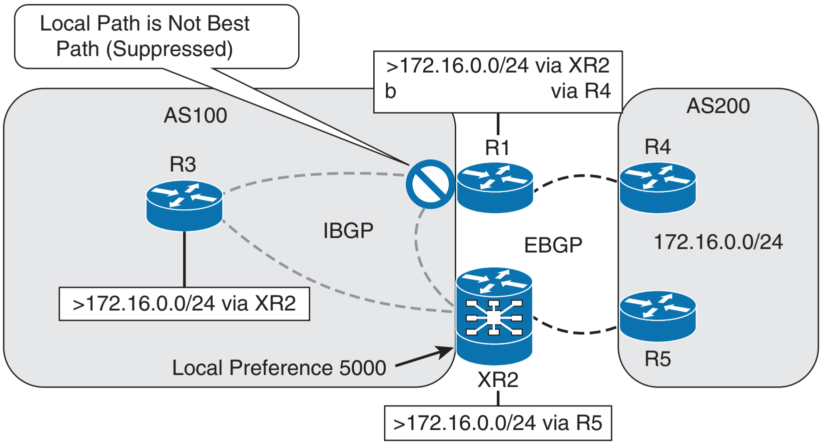

Figure 21-29 *No Secondary Path Information*

The IOS BGP address family command **bgp advertise-best-external** feature may be used by R1 to ensure that the additional external path is still advertised to the internal neighbor routers. The feature ensures that the external best paths are still advertised internally even when the iBGP neighbors have advertised a better internal path to reach the destination prefix.

In IOS XR, the address family command **advertise best-external** ensures the best external path is still advertised internally when a better iBGP route is known.

BGP PIC is automatically enabled on the router when BGP **advertise-best-external** is configured.

BGP Additional Path

One of the design challenges engineers face with BGP PIC is having the necessary alternate paths in the BGP table for precomputing backup repair paths. The regular BGP best path algorithm dictates that only the best path for a prefix is advertised to neighbor routers, so it is not uncommon for internal routers to have only one path for a prefix.

Figure 21-30 demonstrates how a route reflector can inadvertently hide alternate path information by advertising only the best path to clients.

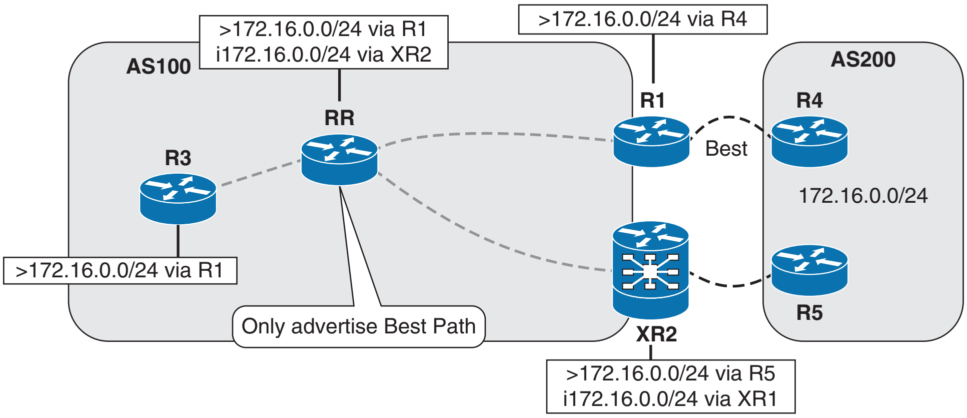

Figure 21-30 *Route Reflector Suppresses Secondary Path Information*

The BGP additional paths feature allows the advertisement of multiple paths and can be deployed along with BGP PIC to ensure that an alternate precomputed path is available for fast reroute.

> **Note** The BGP additional path feature increases memory utilization because the additional backup path increases the BGP table size.

There are three steps for enabling BGP additional path:

Step 1. Enable add path capabilities.

BGP additional paths capability is negotiated between neighbors. A neighbor may send additional paths, receive additional paths, or both. The additional path for all neighbors is enabled in the BGP address family configuration mode:

IOS: **bgp additional-paths** {**send** [**receive**] | **receive**}

IOS XR: **additional-paths receive**, **additional-paths send**

The capability may also be enabled or disabled for individual neighbors:

IOS: **neighbor** *ip-address* **additional-paths disable**

IOS XR: **capability additional-paths send disable**

Step 2. **Identify candidate paths for advertisement.**

In IOS, three candidate selection policies are available for identifying paths for advertisement:

- **bgp additional-paths selec**t best *number*: The best 2 or best 3 command option means that the best path along with the second or second and third best paths are advertised to the neighbor router.
- **bgp additional-paths select group-best:** The **group-best** option chooses the best path option from among the same autonomous system. So, if there are multiple paths to reach a prefix from an autonomous system, only one path is selected. If there are multiple paths to reach the prefix from two neighboring autonomous systems, two best paths are chosen, one best path for each autonomous system.
- **bgp additional-paths select all:** All paths with a unique next-hop forwarding address are advertised to the neighbor.

In IOS XR, the address family command **additional-paths selection route-policy** *route-policy-name* advertises the additional paths. The route policy may match a specific prefix or all prefixes based on the match criteria. Within the RPL, the add path capability is set with the command **set path-selection** {**backup** *number* **| group-best | all | best-path**} [**install**] [**multipath-protect**] [**advertise**]. IOS XR only allows the advertisement of the best and second best path for a prefix.

The RPL policy set condition has the following selection criteria:

- **Backup:** The second best path
- **Group best:** The best path for each neighboring autonomous system
- **All:** All BGP paths
- **Best-path:** Only the best path

Based on the selection, the router can either install a backup path, advertise the additional prefix, or both:

- **Install:** Installs backup path (PIC)
- **Multipath protect:** Install backup path for a multipath (PIC) and advertise
- **Advertise:** Advertise the additional path

The following example demonstrates how to install a backup path for PIC if the local community of the prefix is 1:1. If the prefix is either 172.16.0.0/16 or 172.22.0.0/16, a backup path is installed, and the best and second best backup path for the prefix are advertised to the neighbor routers.

```
IOS XR
route-policy BACKUP
  if community matches-any (1:1) then
    set path-selection backup 1 install
  elseif destination in (172.16.0.0/16, 172.22.0.0/16) then
    set path-selection backup 1 advertise
    set path-selection backup 1 install
  endif
end-policy
```

Step 3. **Advertise paths to neighbors.**

The final step is to activate the additional path selection policy for each neighbor with the IOS command **neighbor** *ip-address* **advertise additional-paths** [**best** *number*] [**group-best**] [**all**]. IOS XR routers do not require Step 3.

Figure 21-31 illustrates the same network, but with the additional path feature enabled. With add path feature enabled, every router now has a backup path to reach the prefix 172.16.0.0/24 in AS200.

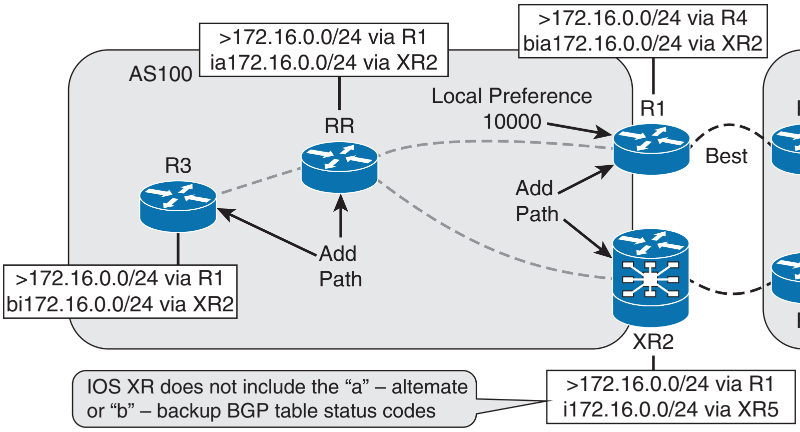

Figure 21-31 *Full PIC protection with the Help of BGP Add Path*

Note The IOS BGP table includes special status codes to indicate that a PIC *b* backup path is installed or an *a* additional path is advertised.

Example 21-50 demonstrates the BGP add path configuration for every router in AS100. In the topology, every iBGP router is configured to support the additional path capability. The route reflector is not part of the data forwarding path, so it has not been configured to install a BGP PIC backup path.

Example 21-50 *BGP PIC Edge and Additional Path Configuration*

```
R1
! Output omitted for brevity
router bgp 100
 neighbor 10.0.1.2 remote-as 200
 neighbor 192.168.100.100 remote-as 100
 neighbor 192.168.100.100 update-source Loopback0
 !
 address-family ipv4
  bgp additional-paths select all
  bgp additional-paths send receive
  bgp additional-paths install
  bgp recursion host
  neighbor 10.0.1.2 activate
  neighbor 192.168.100.100 activate
  neighbor 192.168.100.100 next-hop-self
  neighbor 192.168.100.100 advertise additional-paths all
 exit-address-family
```

```
XR2
! Output omitted for brevity

route-policy BACKUP
  set path-selection backup 1 advertise install
end-policy
!
router bgp 100
 address-family ipv4 unicast
  additional-paths receive
  additional-paths send
  additional-paths selection route-policy BACKUP
  nexthop resolution prefix-length minimum 32
!
 neighbor 10.0.3.5
  remote-as 200
  address-family ipv4 unicast
   route-policy PASS in
   route-policy PASS out
  !
 !
 neighbor 192.168.100.100
  remote-as 100
  update-source Loopback0
  address-family ipv4 unicast
```

```
   next-hop-self
  !
 !
```

R3

```
! Output omitted for brevity
router bgp 100
 neighbor 192.168.100.100 remote-as 100
 neighbor 192.168.100.100 update-source Loopback0
 !
 address-family ipv4
  bgp additional-paths select all
  bgp additional-paths send receive
  bgp additional-paths install
  bgp recursion host
  neighbor 192.168.100.100 activate
  neighbor 192.168.100.100 additional-paths receive
 exit-address-family
```

RR

```
! Output omitted for brevity
interface Loopback0
 ip address 192.168.100.100 255.255.255.255
!
router bgp 100
 neighbor CLIENT peer-group
 neighbor CLIENT remote-as 100
 neighbor CLIENT update-source Loopback0
 neighbor 192.168.1.1 peer-group CLIENT
 neighbor 192.168.2.2 peer-group CLIENT
 neighbor 192.168.3.3 peer-group CLIENT
 !
 address-family ipv4
  bgp additional-paths select all best 2
  bgp additional-paths send receive
  neighbor CLIENT route-reflector-client
  neighbor CLIENT advertise additional-paths best 2
  neighbor 192.168.1.1 activate
  neighbor 192.168.2.2 activate
  neighbor 192.168.3.3 activate
 exit-address-family
```

Example 21-51 demonstrates how to view the BGP neighbor relationship to confirm that the additional path capability has successfully negotiated.

Example 21-51 *Confirming BGP Add Path Capability Exchange*

```
R4-RR#show bgp ipv4 unicast neighbors 192.168.100.100
! Output omitted for brevity
 For address family: IPv4 Unicast
  Additional Paths send capability: advertised and received
  Additional Paths receive capability: advertised and received
```

```
RP/0/0/CPU0:XR3#show bgp ipv4 unicast neighbors 192.168.100.100
! Output omitted for brevity
  AF-dependent capabilities:
    Additional-paths Send: advertised and received
    Additional-paths Receive: advertised and received
    Additional-paths operation: Send and Receive
```

Example 21-52 demonstrates how to view which paths are advertised to the neighbor routers. In the example, the route reflector is advertising two paths for the prefix 172.16.0.0/16, while the edge router XR2 is advertising the local eBGP path.

Example 21-52 *Viewing Additional Path Prefix*

```
RR#show bgp ipv4 unicast 172.16.0.0/24
BGP routing table entry for 172.16.0.0/24, version 76
Paths: (2 available, best #2, table default)
  Path advertised to update-groups:
     4
  Refresh Epoch 1
  200, (Received from a RR-client)
    192.168.3.3 (metric 3) from 192.168.3.3 (192.168.3.3)
      Origin IGP, metric 0, localpref 100, valid, internal, best2, all
      rx pathid: 0x1, tx pathid: 0x1
  Path advertised to update-groups:
     4
  Refresh Epoch 2
  200, (Received from a RR-client)
    192.168.1.1 (metric 3) from 192.168.1.1 (192.168.1.1)
      Origin IGP, metric 0, localpref 10000, valid, internal, best
      rx pathid: 0x0, tx pathid: 0x0
```

```
RP/0/0/CPU0:XR2#show bgp ipv4 unicast 172.16.0.0/24
BGP routing table entry for 172.16.0.0/24
Versions:
  Process           bRIB/RIB  SendTblVer
  Speaker                 51          51
Paths: (2 available, best #2)
  Not advertised to any peer
  Path #1: Received by speaker 0
  Advertised to peers (in unique update groups):
    192.168.100.100
  200
    10.0.3.5 from 10.0.3.5 (172.16.2.1)
      Origin IGP, metric 0, localpref 100, valid, external, backup, add-path
      Received Path ID 0, Local Path ID 3, version 51
      Origin-AS validity: not-found
  Path #2: Received by speaker 0
  Not advertised to any peer
  200
    192.168.1.1 (metric 3) from 192.168.100.100    (192.168.1.1)
      Origin IGP, metric 0, localpref 10000, valid, internal, best, group-best,
import-candidate
      Received Path ID 0, Local Path ID 1, version 44
      Originator: 192.168.1.1, Cluster list: 192.168.100.100
```

Summary

The aim of high availability is to achieve continuous network uptime by designing a network to avoid single points of failure, incorporate deterministic network patterns, and use event-driven failure detection to provide fast network convergence. The high availability features outlined in this chapter can minimize the impact of a network failure.

The following features prevent packet loss by routing through a failure or proactively routing around a failure:

- NSF and NSR separate the control plane from the forwarding plane, allowing traffic to continue to flow even if the RP experiences a hardware or software failure.
- RFD suppresses unstable routes and can be used to avoid sending traffic over sections of the network that have a recent history of unreliability.

The features that can accelerate network routing convergence are as follows:

- Event-driven failure detection, such as carrier delay and BFD, may be used in place of routing protocol keepalive timers to accelerate failure detection from tens of seconds to milliseconds.
- The link-state routing convergence time can be optimized from 1 to 5 seconds to under 1 second through aggressive LSP/LSA propagation and SPF process tuning.
- BGP convergence can be improved through fast peering deactivation, next-hop tracking, accelerated advertisement updates, and tuning the underlying TCP protocol for faster update exchanges. A precomputed backup path provides a fast reroute forwarding option for packets while waiting for the routing protocols to converge around a network failure.

- Loop-free avoidance and BGP PIC provide fast reroute capabilities by calculating a backup route before the failure occurs. When a failure is detected, traffic is quickly redirected to the repair path within tens of milliseconds.

References in This Chapter

Cisco. Cisco IOS Software Configuration Guides, http://www.cisco.com

Cisco. Cisco IOS XR Software Configuration Guides, http://www.cisco.com

Böhmer, Oliver. “Deploying BGP Fast Convergence / BGP PIC” (presented at Cisco Live, London, 2013).

Luc De Ghein. “IP LFA (Loop-Free Alternative): Architecture and Troubleshooting” (presented at Cisco Live, Milan, 2014).

Pete Lumbis. “Routed Fast Convergence” (presented at Cisco Live, San Francisco, 2014).

White, Slice, Retana, *Optimal Routing Design*. Indianapolis: Cisco Press, 2005.

Moy, J., P. Pillay-Esnault, and A. Lindem. RFC 3623, *Graceful OSPF Restart*. IETF, http://www.ietf.org/rfc/rfc3623.txt, November 2003

Nguyen, L., A. Roy, and A. Zinin. RFC 4811, *OSPF Out-of-Band Link State Database (LSDB) Resynchronization*. IETF, http://www.ietf.org/rfc/rfc4811.txt, March 2007

Nguyen, L., A. Roy, and A. Zinin. RFC 4812, *OSPF Restart Signaling*. IETF, http://www.ietf.org/rfc/rfc4812.txt, March 2007

Zinin, A., A. Roy, L. Nguyen, B. Friedman, and D. Yeung. RFC 5613, *OSPF Link-Local Signaling*. IETF, http://www.ietf.org/rfc/rfc4812.txt, August 2009

Shand, M. and L. Ginsberg. RFC 3847, *Restart Signaling for Intermediate System to Intermediate System (IS-IS)*. IETF, http://www.ietf.org/rfc/rfc3847.txt, July 2004

Sangli, S., E. Chen, R. Fernando, J. Scudder, and Y. Rekhter. RFC 4724, *Graceful Restart Mechanism for BGP*. IETF, http://www.ietf.org/rfc/rfc4724.txt, January 2007

Katz, D. and D. Ward. RFC 5880, *Bidirectional Forwarding Detection (BFD)*. IETF, http://www.ietf.org/rfc/rfc5881.txt, June 2010

Katz, D. and D. Ward. RFC 5881, *Bidirectional Forwarding Detection (BFD) for IPv4 and IPv6 (Single Hop)*. IETF, http://www.ietf.org/rfc/rfc5881.txt, June 2010

Katz, D. and D. Ward. RFC 5882, *Generic Application of Bidirectional Forwarding Detection (BFD)*. IETF, http://www.ietf.org/rfc/rfc5882.txt, June 2010

Katz, D. and D. Ward. RFC 5883, *Bidirectional Forwarding Detection (BFD) for Multihop Paths*. IETF, http://www.ietf.org/rfc/rfc5883.txt, June 2010

RFC 2328, *OSPF Version 2*, J. Moy, IETF, http://www.ietf.org/rfc/rfc2328.txt, April 1998

Rao, S., A. Zinin, and A. Roy. RFC 3883, *Detecting Inactive Neighbors over OSPF Demand Circuits (DC)*. IETF, http://www.ietf.org/rfc/rfc3883.txt, October 2004

Atlas, A. and A. Zinin. RFC 5286, *Basic Specification for IP Fast Reroute: Loop-Free Alternates*. IETF, http://www.ietf.org/rfc/rfc5286.txt, September 2008

Filsfil, C. and P. Francois. RFC 6571, *Loop-Free Alternate (LFA) Applicability in Service Provider (SP) Networks.* IETF, http://www.ietf.org/rfc/rfc6571.txt, June 2012