Configure and Verify the BGP Conditional Advertisement Feature

Contents

Introduction

Prerequisites

- Requirements
- Components Used

Background Information

Configure

- Network Diagram
- Configure the Conditional Advertisement Feature

Verify the Configuration

- Example 1
- Example 2
- Example 3

Related Information

Introduction

This document describes how to configure and verify the Border Gateway Protocol (BGP) conditional advertisement feature.

Prerequisites

Requirements

Cisco recommends that you have knowledge of this topic:

- BGP Case Studies

Components Used

This document is not restricted to specific software and hardware versions.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Background Information

The Border Gateway Protocol (BGP) conditional advertisement feature provides additional control of route advertisement, depending on the existence of other prefixes in the BGP table.

The BGP conditional advertisement feature described in this document was introduced in Cisco IOS® Software Release 11.1 and 11.2 and is available in later releases.

Normally, routes are propagated regardless of the existence of a different path. The BGP conditional advertisement feature uses the non-exist-map and the advertise-map keywords of the **neighbor advertise-map** command in order to track routes by the route prefix. If a route prefix is not present in output of the **non-exist-map** command, then the route specified by the **advertise-map** command is announced. This feature is useful for multihomed networks, in which some prefixes are advertised to one of the providers only if information from the other provider is not present (this indicates a failure in the peering session or partial reachability).

The conditional BGP announcements are sent in addition to the normal announcements that a BGP router sends to its peers.

Configure

In this section, you are presented with the information to configure the features described in this document.

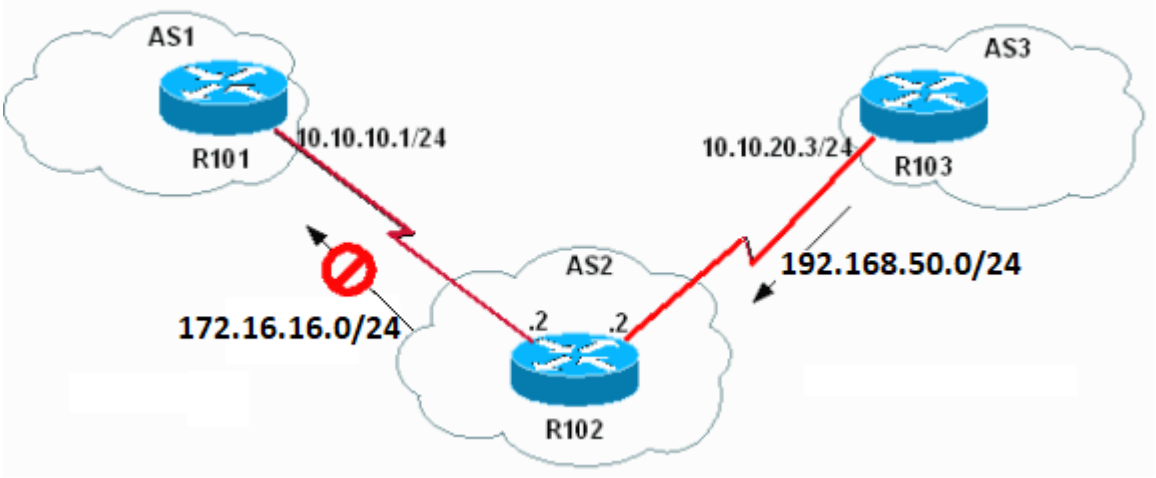


Note: To find additional information on the commands used in this document, use the Cisco IOS Command Lookup tool. Only registered Cisco users can access internal Cisco tools and information.

Network Diagram

This document uses the network setup shown in this diagram.

Here, the loopback interface of R103 is used to announce 192.168.50.0/24 to R102. R102 has a BGP network, 172.16.16.0/24, which is announced to both its external BGP (eBGP) peers, R101 and R103, by default.



Network Setup

With the BGP conditional advertisement feature, you can now accomplish these tasks on R102:

- If 192.168.50.0/24 exists in R102's BGP table, do not advertise the 172.16.16.0/24 network to R101.
- If 192.168.50.0/24 does not exist in R102's BGP table, advertise the 172.16.16.0/24 network to R101.

With Cisco IOS 12.2T or later versions, the exist-map keyword can be used to accomplish these tasks as well:

- If 192.168.50.0/24 exists in the R102 BGP table, advertise the 172.16.16.0/24 network to R101.
- If 192.168.50.0/24 does not exist in the R102 BGP table, do not advertise the 172.16.16.0/24 network to R101.

Configure the Conditional Advertisement Feature

This document uses these configurations:

Note: The example here discusses the non-exist-map keyword. Usage of the exist-map keyword is similar to this.

R102
<#root> hostname R102 ! interface Loopback0 ip address 172.16.16.1 255.255.255.0 ! interface Serial8/0

```
 ip address 10.10.10.2 255.255.255.0
!
interface Serial9/0
 ip address 10.10.20.2 255.255.255.0
!
router bgp 2
 bgp log-neighbor-changes
 network 172.16.16.0 mask 255.255.255.0
 network 172.31.130.0
 neighbor 10.10.10.1 remote-as 1
 neighbor 10.10.10.1 advertise-map ADVERTISE non-exist-map NON-EXIST


!--- Advertises the routes matched in the route-map ADVERTISE (172.16.16.0/24)
!--- only if the routes matched in route-map NON-EXIST (192.168.50.0/24)
!--- do not exist in the BGP table.

 neighbor 10.10.20.3 remote-as 3
!
ip route 172.31.130.0 255.255.0.0 Null0
!
access-list 60 permit 172.16.16.0 0.0.0.255
access-list 65 permit 192.168.50.0 0.0.0.255
!
route-map NON-EXIST permit 10
 match ip address 65


!

route-map ADVERTISE permit 10
 match ip address 60


!
```

R103

```
hostname R103
!
interface Loopback0
 ip address 192.168.50.1 255.255.255.0
!
interface Serial9/0
 ip address 10.10.20.3 255.255.255.0
!
router bgp 3
 bgp log-neighbor-changes
 network 192.168.50.0
 neighbor 10.10.20.2 remote-as 2
!
```

R101

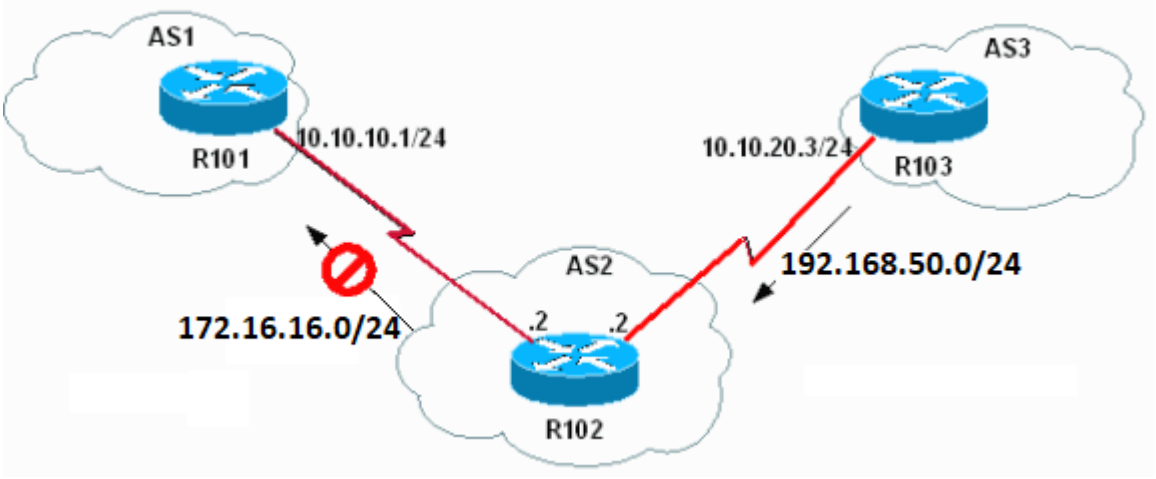
```
hostname R101
!
interface Loopback0
```

```
 ip address 10.200.200.1 255.255.255.0
!
interface Serial8/0
 ip address 10.10.10.1 255.255.255.0
!
router bgp 1
 bgp log-neighbor-changes
 network 10.200.200.0
 neighbor 10.10.10.2 remote-as 2
!
```

Verify the Configuration

Example 1

This example verifies what BGP looks like when 192.168.50.0/24 is in the R102 BGP table:



Verify the Configuration

First, check to see if 192.168.50.0/24 exists in the R102 BGP table:

<#root>

R102#

show ip bgp

```
BGP table version is 6, local router ID is 172.16.16.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 172.16.16.0/24   0.0.0.0                  0         32768 i
*> 172.31.130.0     0.0.0.0                  0         32768 i
```

```
*> 192.168.50.0     10.10.20.3               0             0 3 i

*> 10.200.200.0     10.10.10.1               0             0 1 i
```

Since 192.168.50.0/24 is in the R102 BGP table, R102 must not advertise 172.16.16.0/24 to R101.

```
<#root>

R102#

show ip bgp neighbors 10.10.10.1 advertised-routes


BGP table version is 6, local router ID is 172.16.16.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 172.31.130.0      0.0.0.0                  0         32768 i
*> 192.168.50.0     10.10.20.3               0             0 3 i

!--- Note 172.16.16.0/24 is not advertised to neighbor 10.10.10.1.


R102#

show ip bgp 172.16.16.0

BGP routing table entry for 172.16.16.0/24, version 6
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:

!--- This is not advertised to R101.

10.10.20.3
  Local
    0.0.0.0 from 0.0.0.0 (172.16.16.1)
      Origin IGP, metric 0, localpref 100, weight 32768, valid, sourced, local, best
```

Then check the status of the conditional advertisement on R102:

```
<#root>

R102#

show ip bgp neighbors 10.10.10.1

BGP neighbor is 10.10.10.1,  remote AS 1, external link
  BGP version 4, remote router ID 10.200.200.1
  BGP state = Established, up for 02:27:07
  Last read 00:00:07, hold time is 180, keepalive interval is 60 seconds
```

```
!--- Output suppressed.


 For address family: IPv4 Unicast
  BGP table version 6, neighbor version 6
  Index 1, Offset 0, Mask 0x2
  Condition-map NON-EXIST, Advertise-map ADVERTISE, status: Withdraw
  1 accepted prefixes consume 36 bytes
  Prefix advertised 3, suppressed 0, withdrawn 1
  Number of NLRIs in the update sent: max 1, min 0


!--- Output suppressed.
```

The output shows that the conditional advertisement is withdrawn and that the networks that match the route-map ADVERTISE are not announced to the peer 10.10.10.1.

In order to confirm the routes that match the route-map ADVERTISE are not announced to R101, check the BGP table on R101:

```
<#root>

R101#

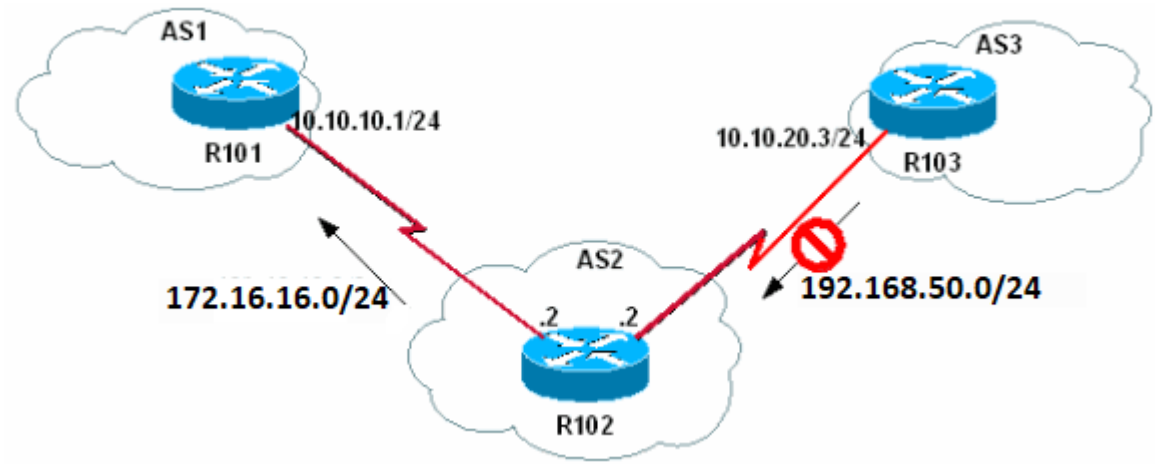
show ip bgp 172.16.16.0

%

Network not in table
```

Example 2

This example shows what BGP looks like when network 192.168.50.0/24 does not exist in the R102 BGP table:



192.168.50.0/24 Does not Exist in the R102 BGP Table

The steps are:

1. Shut down the loopback interface 0 on R103 so that R103 no longer announces 192.168.50.0/24 to R102.

```
<#root>

R103(config)#

interface loopback 0

R103(config-if)#

shutdown


R103(config-if)#
03:29:36: %LINK-5-CHANGED: Interface Loopback0, changed state to administratively down
```

2. Verify that R102 does not learn 192.168.50.0/24 and that the network is not included in the R102 BGP table.

```
<#root>

R102#

show ip bgp


BGP table version is 8, local router ID is 172.16.16.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 172.16.16.0/24   0.0.0.0                  0         32768 i
```

```
*> 172.31.130.0     0.0.0.0                  0         32768 i
*> 10.200.200.0     10.10.10.1               0         0 1 i

!--- Note 192.168.50.0/24 is not present.
```

3. Observe how long it takes for the conditional advertisement to start:

```
<#root>

R102#

debug ip bgp updates

*Mar  1 02:39:18.059: BGP(0):

10.10.20.3 rcv UPDATE about 192.168.50.0/24

-- withdrawn

*Mar  1 02:39:18.059: BGP(0): no valid path for 192.168.50.0/24
*Mar  1 02:39:18.079: BGP(0): nettable_walker 192.168.50.0/24 no best path
*Mar  1 02:39:18.219: BGP(0): 10.10.10.1 computing updates, afi 0, neighbor
version 10, table version 11, starting at 0.0.0.0
*Mar  1 02:39:18.219: BGP(0): 10.10.10.1 send unreachable 192.168.50.0/24
*Mar  1 02:39:18.219: BGP(0):

10.10.10.1 send UPDATE 192.168.50.0/24

-- unreachable

*Mar  1 02:39:18.219: BGP(0): 10.10.10.1 1 updates enqueued (average=27, maximum=27)
*Mar  1 02:39:18.219: BGP(0): 10.10.10.1 update run completed, afi 0, ran for 0ms,
neighbor version 10, start version 11, throttled to 11

*Mar  1 02:40:04.747: BPG(0):

Condition NON-EXIST changes to Advertise

*Mar  1 02:40:04.747: BGP(0):

net 172.16.16.0/24 matches ADV MAP ADVERTISE:


bump version to 12

*Mar  1 02:40:05.187: BGP(0): nettable_walker 172.16.16.0/24 route sourced
locally
*Mar  1 02:40:05.187: BGP(0): 10.10.10.1 computing updates, afi 0, neighbor
version 11, table version 12, starting at 0.0.0.0
*Mar  1 02:40:05.187: BGP(0):

10.10.10.1 172.16.16.0/24 matches advertise map

ADVERTISE, state: Advertise

*Mar  1 02:40:05.187: BGP(0):

10.10.10.1 send UPDATE (format) 172.16.16.0/24,
```

```
next 10.10.10.2, metric 0, path


*Mar  1 02:40:05.187: BGP(0): 10.10.10.1 1 updates enqueued (average=52, maximum=52)
*Mar  1 02:40:05.187: BGP(0): 10.10.10.1 update run completed, afi 0, ran for 0ms,
neighbor version 11, start version 12, throttled to 12
```



Note: The debugging output can differ a bit depending on the software version used.

The conditional advertisement process is triggered by the BGP scanner process, which runs every 60 seconds. This means that the maximum time for the conditional advertisement to take effect is 60 seconds. The conditional advertisement can take effect sooner, depending on when the tracked route is removed from the BGP table and when the next instance of the BGP scanner occurs.

Run these commands on R102 in order to verify the conditional advertisement state on R102 for the neighbor 10.10.10.1:

```
<#root>

R102#

show ip bgp neighbors 10.10.10.1

BGP neighbor is 10.10.10.1,  remote AS 1, external link
  BGP version 4, remote router ID 10.200.200.1
  BGP state = Established, up for 02:45:27
  Last read 00:00:27, hold time is 180, keepalive interval is 60 seconds

!--- Output suppressed.


 For address family: IPv4 Unicast
  BGP table version 12, neighbor version 12
  Index 1, Offset 0, Mask 0x2

  Condition-map NON-EXIST, Advertise-map ADVERTISE, status: Advertise

  1 accepted prefixes consume 36 bytes
  Prefix advertised 6, suppressed 0, withdrawn 4
  Number of NLRIs in the update sent: max 1, min 0

!--- Output suppressed.
```

The R101 BGP table and routing table has 172.16.16.0/24, as shown here:

Note: In this sample output, the conditional BGP announcements (network 172.16.16.0/24) are sent, in addition to the normal BGP announcements (network 172.31.130.0/16) that a BGP router sends to its peers.

```
<#root>

R101#

show ip bgp

BGP table version is 18, local router ID is 10.200.200.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 172.16.16.0/24   10.10.10.2               0             0 2 i
*> 172.31.130.0     10.10.10.2               0             0 2 i
*> 10.200.200.0     0.0.0.0                  0         32768 i
```

```
<#root>

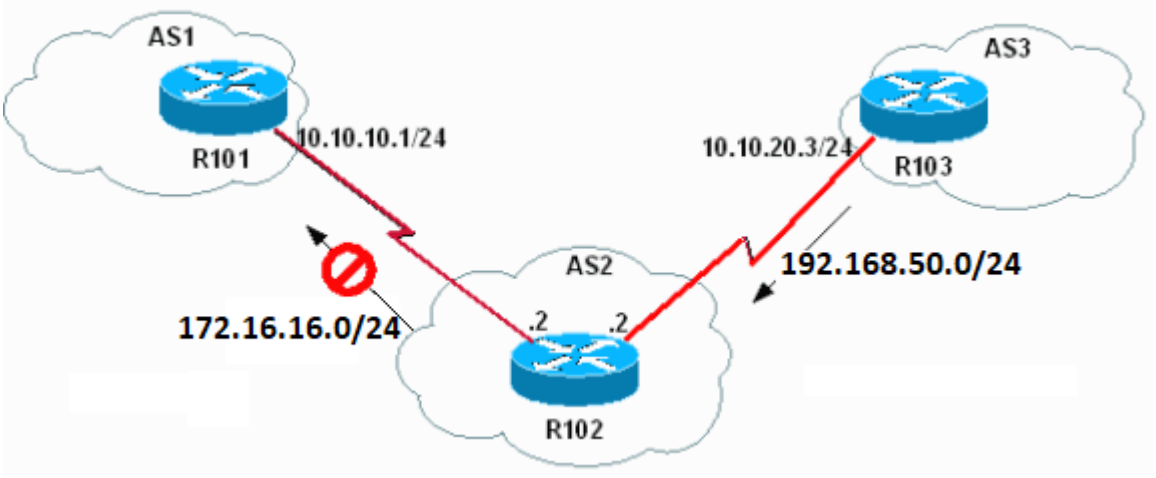
R101#

show ip route bgp

     172.16.0.0/24 is subnetted, 1 subnets
B       172.16.16.0 [20/0] via 10.10.10.2, 00:09:32
B    172.31.130.0/16 [20/0] via 10.10.10.2, 02:48:46
```

Example 3

This example involves the reinstallation of the 192.168.50.0/24 network in R102 in order to see how BGP changes from Advertise to Withdraw .



How BGP Changes from Advertise to Withdraw

In order to reinstall 192.168.50.0/24, run the **no shutdown** command to bring the interface loopback 0 on R103 to **UP**.

```
<#root>

R103(config)#

interface loopback 0

R103(config-if)#

no shutdown

R103(config-if)#
03:49:06: %LINK-3-UPDOWN: Interface Loopback0,
changed state to up
03:49:07: %LINEPROTO-5-UPDOWN: Line protocol on
Interface Loopback0, changed state to up

!--- R102 kicks in conditional advertisement the moment the
!--- conditional network is received again.

*Mar  1 02:51:42.227: BGP(0): 10.10.20.3 rcvd UPDATE w/ attr:
nexthop 10.10.20.3, origin i, metric 0, path 3
*Mar  1 02:51:42.227: BGP(0):

10.10.20.3 rcvd 192.168.50.0/24

*Mar  1 02:51:42.247: BGP(0): Revise route installing 192.168.50.0/24 ->
10.10.20.3 to main IP table
*Mar  1 02:51:42.379: BGP(0): 10.10.10.1 computing updates, afi 0,
neighbor version 12, table version 13, starting at 0.0.0.0
*Mar  1 02:51:42.379: BGP(0):

10.10.10.1 send UPDATE (format)

192.168.50.0/24, next 10.10.10.2, metric 0, path 3

*Mar  1 02:51:42.379: BGP(0): 10.10.10.1 1 updates enqueued
(average=47, maximum=47)
*Mar  1 02:51:42.379: BGP(0): 10.10.10.1 update run completed, afi 0,
ran for 0ms, neighbor version 12, start version 13, throttled to 13
```

```
*Mar  1 02:52:09.159: BPG(0):

Condition NON-EXIST changes to Withdraw

*Mar  1 02:52:09.159: BGP(0): net 172.16.16.0/24 matches ADV MAP
ADVERTISE: bump version to 14
*Mar  1 02:52:09.499: BGP(0): nettable_walker 172.16.16.0/24 route
sourced locally
*Mar  1 02:52:11.559: BGP(0): 10.10.10.1 computing updates, afi 0,
neighbor version 13, table version 14, starting at 0.0.0.0
*Mar  1 02:52:11.559: BGP(0):

10.10.10.1 172.16.16.0/24 matches advertise



map ADVERTISE, state: Withdraw

*Mar  1 02:52:11.559: BGP(0):

10.10.10.1 send unreachable 172.16.16.0/24

*Mar  1 02:52:11.559: BGP(0):

10.10.10.1 send UPDATE 172.16.16.0/24 -- unreachable

*Mar  1 02:52:11.559: BGP(0): 10.10.10.1 1 updates enqueued (average=27, maximum=27)
*Mar  1 02:52:11.559: BGP(0): 10.10.10.1 update run completed, afi 0, ran for 0ms,
neighbor version 13, start version 14, throttled to 14
```

R102 no longer advertises 172.16.16.0/24 to R101.

```
<#root>

R102#

show ip bgp neighbors 10.10.10.1 advertised-routes

BGP table version is 14, local router ID is 172.16.16.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 172.31.130.0     0.0.0.0                  0         32768 i
*> 192.168.50.0     10.10.20.3               0             0 3 i

!--- Note 172.16.16.0/24 is not advertised.

R102#

show ip bgp neighbors 10.10.10.1

BGP neighbor is 10.10.10.1,  remote AS 1, external link
  BGP version 4, remote router ID 10.200.200.1
  BGP state = Established, up for 03:01:32
  Last read 00:00:31, hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
!--- Output supressed.
```

```
 For address family: IPv4 Unicast
  BGP table version 14, neighbor version 14
  Index 1, Offset 0, Mask 0x2

  Condition-map NON-EXIST, Advertise-map ADVERTISE, status: Withdraw

  1 accepted prefixes consume 36 bytes
  Prefix advertised 7, suppressed 0, withdrawn 5
  Number of NLRIs in the update sent: max 1, min 0

!--- Output supressed.
```



Note: In a real world scenario, AS1 and AS3 are connected to the Internet (Global BGP). Hence the network 192.168.50.0/24 could be propagated through a global bgp mesh from AS3 to the AS1 (R101) BGP table. AS1, in turn, could propagate the prefix 192.168.50.0 to R102 (depending on the policy agreement between AS1 and AS2). If R101 propagates AS3 routes learned from the global BGP mesh to R102, then the conditional advertisement can fail if additional checks are not placed in the non-exist map.

In order to better understand why conditional advertisement fails, consider this scenario. AS1 learns 192.168.50.0/24 from the global bgp mesh and announces 192.168.50.0/24 to AS2 (R102). R102 also learns the prefix 192.168.50.0/24 from its direct peering to R103 (over the R102 to R103 link). When the direct link between R102 and R103 fails, you expect that the prefix 192.168.50.0/24 ceases to exist in R102's BGP table and the conditional advertisement kicks in and announces prefix 172.16.16.0/24 to R101. However, since the prefix 192.168.50.0/24 continues to exist in the R102 BGP table (learned from R101), the conditional advertisement breaks because the prefix in the non-exist map still exists in the R102 BGP table.

To ensure that the prefix 192.168.50.0/24 is learned only from the direct connection to AS3 (R102 to R103) in order for conditional advertisement to kick in, add a **match as_path** statement under the **non-exist map** that matches the AS_PATH of prefix 192.168.50.0/24 learned from direct connection (which in this case is AS 3). The regular expression for this case is ^3 . For more information on regular expressions, refer to Using Regular Expressions in BGP. Beware that you cannot match just the as-path, and not match the prefix. The match as-path can only supplement the matching criteria for the matched prefix. In other words, you cannot advertise some prefixes to a neighbor if no prefix exists from a certain AS #.

The new configuration on R102 is shown here. The additions are in bold.

R101
<#root> hostname R102 ! interface Loopback0 ip address 172.16.16.1 255.255.255.0 ! interface Serial8/0 ip address 10.10.10.2 255.255.255.0 !

```
interface Serial9/0
 ip address 10.10.20.2 255.255.255.0
!
router bgp 2
 bgp log-neighbor-changes
 network 172.16.16.0 mask 255.255.255.0
 network 172.31.130.0
 neighbor 10.10.10.1 remote-as 1
 neighbor 10.10.10.1 advertise-map ADVERTISE non-exist-map NON-EXIST
 neighbor 10.10.20.3 remote-as 3
!
ip route 172.31.130.0 255.255.0.0 Null0
!
ip as-path access-list 1 permit ^3


!
access-list 60 permit 172.16.16.0 0.0.0.255
access-list 65 permit 192.168.50.0 0.0.0.255
!
route-map NON-EXIST permit 10
 match ip address 65


match as-path 1


!
route-map ADVERTISE permit 10
 match ip address 60
!
```

Related Information

- BGP: Frequently Asked Questions
- Examine Border Gateway Protocol Case Studies
- Cisco Technical Support & Downloads