Allied Telesis

Configuring IPv6 Routing Solutions using BGP4+

Version 1.0

Compiled by

Jason Higgins and Sheldon Duthie

Configuring IPv6 Routing Solutions using BGP4+

First Edition

Compiled by:

Jason Higgins and Sheldon Duthie

About Allied Telesis, Inc.

Founded in 1987, Allied Telesis is a world class leader in delivering IP/Ethernet network solutions to the global marketplace. We create innovative, standards-based IP networks that seamlessly connect users with their voice, video and data services. We are an international company headquartered in Japan with major divisions in Europe, Asia and North and South America. Our partners include the world's largest distributors, integrators, solution providers and resellers to assure you receive immediate local service and support. Allied Telesis has been designing, manufacturing and selling networking products for over 20 years.

Our philosophy of producing products of the highest quality, at affordable prices, has resulted in Allied Telesis products being deployed in networks of all types and sizes across the world. Our proven track record of providing solid technology, excellent support and full feature products has allowed Allied Telesis to become the worldwide de-facto standard in many areas of technology. With a portfolio of products that can provide end-to-end networking for both Service Provider, Enterprise and SMB customers, Allied Telesis is the natural choice for many world class organizations.

Please visit us online at **alliedtelesis**.com

Trademarks

Allied Telesis, AlliedWare Plus, EPSRing, SwitchBlade, and VCStack are trademarks or registered trademarks in the United States and elsewhere of Allied Telesis, Inc. Adobe, Acrobat, and Reader are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries. Additional brands, names, and products mentioned herein may be trademarks of their respective companies.

Warning and Disclaimer

The information in this guide is provided on an "as is" basis. The author and the publishers shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this guide.

Produced in New Zealand
First Printing: March 2014
Document Number:
C613-04008-00-REV A

ISBN: 978-0-473-26986-9

Co-authors
Jason Higgins
Sheldon Duthie

Technical Writer
Carin van Bolderen

Illustrator
Angela Maxwell

Editor
Carin van Bolderen

Review Team
Andrew Riddell
Sheldon Duthie
Miranda de Gouw

Cover Designer
Angela Devonport

About this Guide

This guide briefly describes the Border Gateway Protocol 4+(BGP4+) in general terms and how to configure a variety of IPv6 routing scenarios using BGP4+ on AlliedWare Plus™. This guide is intended for audiences with a good understanding of network routing protocols and device configuration.

The discussion begins with an overview of the BGP4+ protocol and related RFCs, how it operates, and where it is normally implemented. This is followed by a basic set-up guide which also outlines inter-operation with other vendors, basic and advanced set-up instructions for BGP peering, and how to verify BGP settings using appropriate AlliedWare Plus **show** commands. This section rounds off with a description of BGP and BGP4+ terms used throughout the guide. The guide then provides a variety of BGP4+ configuration scenarios. Each scenario includes a simple network diagram, step-by-step configuration instructions, and examples of command output verifying the configuration.

The configuration examples are grouped into the following areas:

Which products, software version and license does it apply to?

These configurations apply to AlliedWare Plus software version **5.4.3-2.5** and above, for the following Allied Telesis switches:

- SwitchBlade x8100 Series
- SwitchBlade x8100 Series with - CFC400, CFC960 controller and Premium license, AlliedWare Plus software version **5.4.4** only
- SwitchBlade x908 extended mode, with Advanced L3 license
- x610 Series switches with Advanced L3 license
- x900 Series switches with Advanced L3 license


BGP4+

Contents

CHAPTER 1

BGP Overview

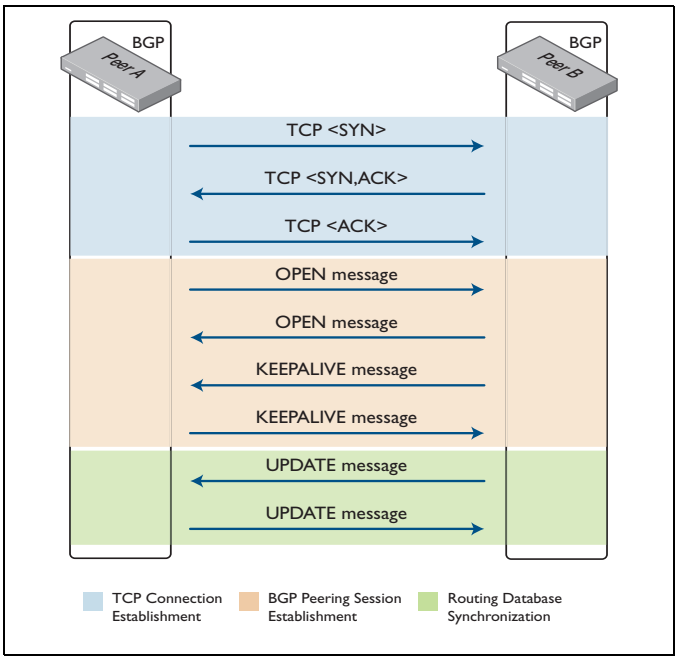
Border Gateway Protocol (BGP for IPv4 and BGP4+ for IPv6) is an exterior gateway protocol (EGP). The purpose of BGP is to advertise, learn, and choose the best paths via the Internet. The BGP4 protocol, defined in RFC 4271, is an external routing protocol used to connect different Autonomous Systems (AS). The BGP4 protocol was updated by RFC 4760 which added support for Multiprotocol Extensions for BGP, including IPv6 address families and other types of addresses. The use of BGP in IPv6 networks is defined in RFC 2545.

The term BGP4+ is used to refer to BGP deployment involving IPv6 networks.

BGP4+ operation

Like BGP, BGP4+ operates at OSI Layer 4 over TCP. Peering sessions are formed between BGP and BGP4+ capable devices, to destination port 179. Once a TCP session has been established, the two BGP devices negotiate the peering by checking the BGP OPEN message.

The OPEN message contains information such as the BGP version, Autonomous System Number (ASN), hold time, and BGP Identifier. Once the peering has been negotiated, BGP scans the Routing Information Base (RIB) and generates UPDATE messages which are sent to the adjacent peer. The UPDATE message contains information such as the BGP attributes and NLRI (Next Layer Reachability Information), which are the prefixes the BGP4+ speaker is advertising. The BGP peering session is maintained with the use of BGP KEEPALIVE messages.



BGP maintains three routing tables:

1. Adj-RIB-IN (Adjacent Routing Information Base Incoming): This table contains unprocessed routing information, received in UPDATE messages from other BGP speakers. By default, the information contained in the Adj-RIB-In is discarded after the routes are processed and selected routes sent to the Loc-RIB. AlliedWare Plus BGP can retain a copy of the Adj-RIB-In in memory using the command: **neighbor {<ipv6-addr> | <peer-group>} soft-reconfiguration inbound**

   - To view the Adj-RIB-In per neighbor, use the command: **show bgp ipv6 neighbors <ipv6-addr> received-routes**

2. Loc-RIB (Local Routing Information Base): This table contains routing information from the Adj-RIBs-In which have been passed through the local inbound routing policy.

   - To view all routes in the Loc-RIB use the command: **show bgp ipv6**
   - To view a specific route in the Loc-RIB use the command: **show bgp ipv6 <ipv6-addr>**.
   - To view a specific set of routes in the Loc-RIB received from a neighbor, use the command: **show bgp ipv6 neighbors <ipv6-addr> routes**

3. Adj-RIB-OUT (Adjacent Routing Information Base Outgoing), NLRI sent to a neighbor. This table contains the routes for advertisement to peers, which have passed through the local outbound routing policy.

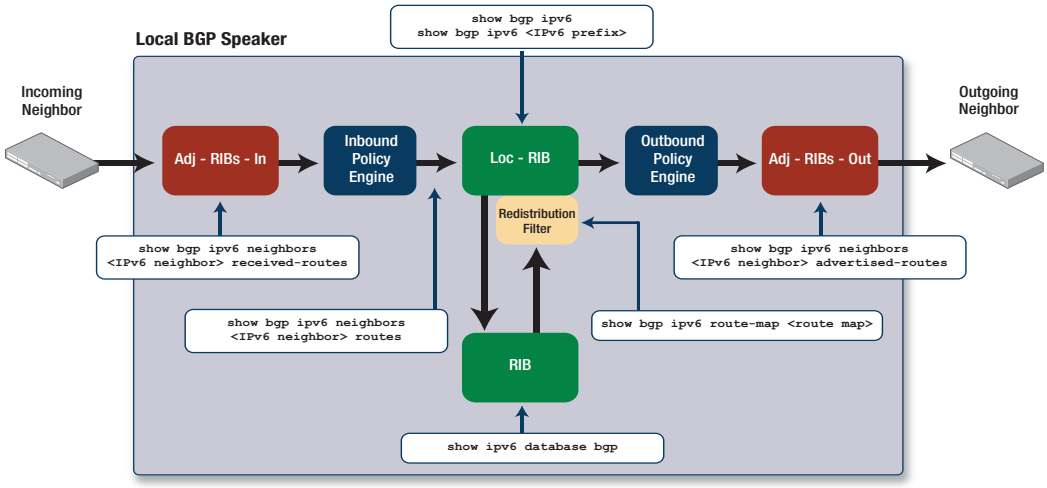
   - To view the Adj-RIB-Out per neighbor, use the command: **show bgp ipv6 neighbors <ipv6-addr> advertised-routes**

The BGP4+ routing protocol features in brief

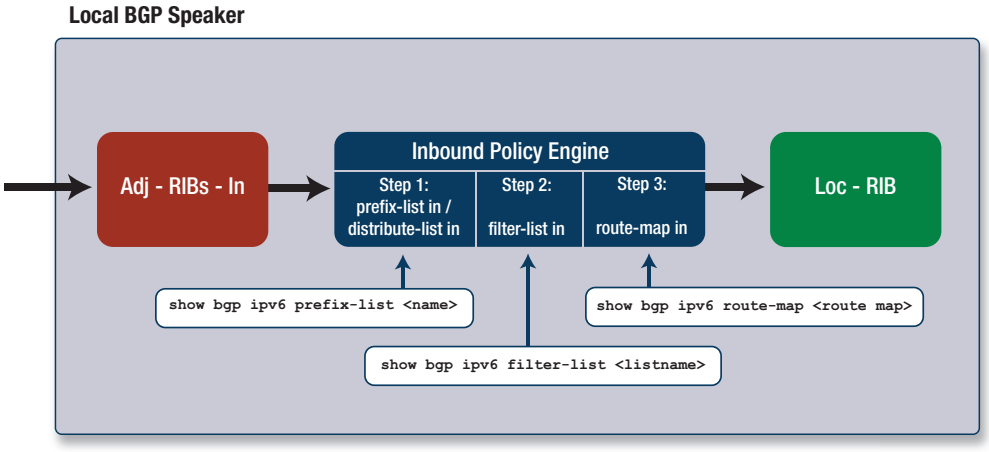
- Prefixes (routes) are based on path and local routing policy information.
- Prefix information consists of the destination, a gateway, and the entire path to the destination.
- Distribution of prefix information is reliable, as exchanges are transported over the TCP protocol between peering devices via TCP port 179.
- There is no dynamic discovery of BGP peers. Peers are statically configured and they do not have to be directly connected to each other via same network segment to be able to peer to each other.
- The decision on which is the best routing path is influenced by filtering policies and prefix selections made by other routers and the BGP best path selection algorithm.

BGP4+ show commands

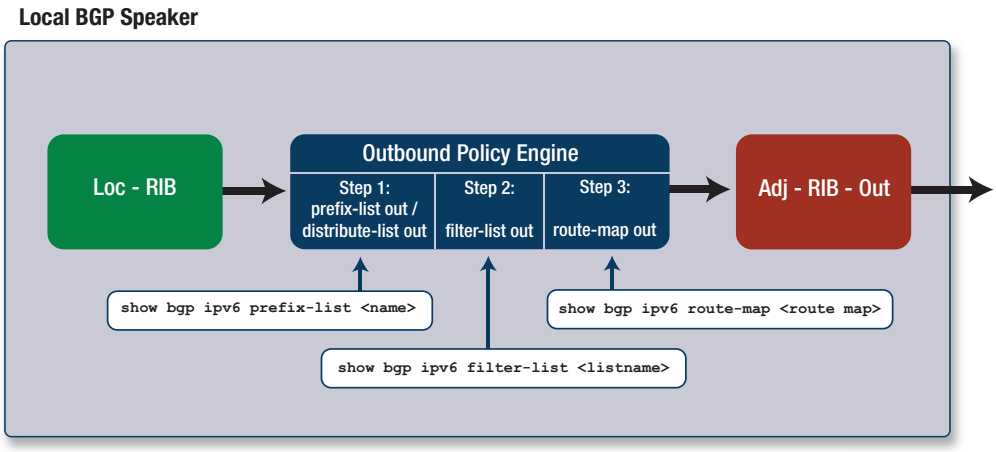
Here's a diagram showing the process of a route coming into/going out of the BGP Loc-RIB. The diagram indicates which show commands to use and when to use them.



This diagram shows the steps and commands relevant to the BGP Loc-RIB **Inbound** policy Engine.



This diagram shows the steps and commands relevant to the BGP Loc-RIB **Outbound** Policy Engine.



Notes on BGP4+ show commands

show bgp ipv6 neighbors <IPv6 addr> received-routes

- Inbound soft reconfiguration MUST be enabled first.
- This command shows ALL routes received from a BGP peer BEFORE filtering occurs. (Contents in the Adj-RIB-In per neighbor.)
- Both accepted and rejected routes are displayed.

show bgp ipv6 neighbors <IPv6 addr> routes

- This command shows routes received from a BGP peer AFTER filtering has occurred. (Contents in the Loc-RIB from a selected neighbor.)
- Only accepted routes are displayed.

show bgp ipv6

- This command shows routes that have been installed in BGP's Loc-RIB (BGP table).

show bgp ipv6 <IPv6 Prefix>

- This command shows detailed information about a prefix installed in the BGP table.

Information includes: Next hop, Attributes, Validity, Best selected route, and Last update.

show bgp ipv6 neighbors <IPv6 addr> advertised-routes

- This command shows routes which have been allowed after filtering and advertised to the specified BGP peer. (Adj-RIB-Out contents per neighbor.)

BGP4+ implementation

BGP4+ can be deployed for routing between different autonomous systems (inter-AS routing), as well as routing within the same autonomous system (intra-AS routing). BGP4+ routers operating in inter-AS routing networks are known as External BGP (eBGP) peers, where as BGP4+ routers operating in intra-AS routing networks are known as Internal BGP (iBGP) peers.

Inter-operation with other vendors

By default AlliedWare Plus configures BGP in extended mode, and follows many settings that are compulsory or recommended in the RFCs.

Extended mode

Using extended mode, BGP4 and BGP4+ communities are allowed to be sent and received by default. This could cause problems in some networks if unauthorized BGP peers are advertising BGP communities in order to adjust routing decisions.

Recommended

- When required, configure the AlliedWare Plus device to use the standard BGP configuration type mode.
- Changing modes requires the device be rebooted.
- When the device reboots, it will load with the standard BGP settings commonly used by most vendors.

```
SW1(config)#bgp config-type standard
SW1(config)#exit
SW1#reload
reboot system? (y/n): y
```

Disabling automatic IPv4 information exchange

- Use the command below to disable the exchange of IPv4 prefixes with a peer. By default BGP will exchange IPv4 prefixes including via IPv6 peering session. It is recommended that this is configured before any neighbors are configured.

```
SW1(config-router)#no bgp default ipv4-unicast
```

AlliedWare Plus timers

- The AS origination interval timer may not be available to adjust on other vendors equipment.
  - If it is desired to maintain common timer policies across the AS, the recommendation is to leave it at the default 15 seconds.
  - Here is an example of restoring the AS origination interval to 15 seconds:

```
SW1(config-router)#no neighbor 2001:db8:10::2 as-origination-interval
```

Other vendor inter-operation

- Some vendors default to using a route advertisement interval of 0 seconds for iBGP peers.
- AlliedWare Plus uses a 5 second route advertisement interval for iBGP peers, as recommended in RFC4271.
- Some vendors may use a 60 second keepalive timer and a 180 second holdtimer.
- AlliedWare Plus uses a 30 second keepalive timer and a 90 second holdtimer.

To maintain consistency across peers you may wish to adjust the timers to match.

Redistribution of BGP routes into OSPF

Some vendors follow RFC 1403 and set the default seed metric to 1 when redistributing BGP routes into OSPFv2 and OSPFv3. The methods used in RFC 1403 are not widely used on the Internet and as such it has been proposed to reclassify the RFC to Historic by RFC 3166.

AlliedWare Plus follows RFC 3166 which means that BGP routes redistributed into OSPF will have a default metric of 20 (the same as all other route types redistributed into OSPF). The following commands can be used to change the default metric for OSPF routes redistributed into BGP.

IPv4

```
awplus(config)#router ospf
awplus(config-router)#redistribute bgp metric 1
```

IPv6

```
awplus(config)#router ipv6 ospf
awplus(config-router)#redistribute bgp metric 1
```

More information:

- http://tools.ietf.org/html/rfc1403
- http://tools.ietf.org/html/rfc3166

BGP and BGP4+ terminology

The table below provides descriptions of BGP and BGP4+ terms used through this guide:

TERM	DESCRIPTION
AS number (ASN)	A number that identifies an AS. Previously, an ASN was a 16-bit number between 1 and 64,511 (public) and 64,512 and 65,535 (private) assigned to an AS for the purpose of BGP operation. Since RFC 6793, an ASN is now 32-bits (4-bytes) with a range of decimal values between 1 and 4,294,967,295.
AS_Path	A BGP path attribute that lists ASNs through which the route has been advertised.
AS_Sequence	A type of AS_Path segment consisting of an ordered list of ASNs through which the route has been advertised.
Auto summary	A routing protocol feature in which a router that connects to more than one classful network advertises summarized routes for each entire classful network when sending updates out interfaces connected to other classful networks.
Autonomous System (AS)	A set of routers inside a single administrative authority, grouped together for the purpose of controlling routing policies for the routes advertised by that group to the Internet.
BGP best path algorithm	The process that BGP uses to examine competing BGP paths (routes) in its BGP table, for a single prefix, choosing one route as the best route. The best path algorithm results in a choice of a single route for each prefix as that router's best BGP path. BGP compares the current best path against the next path in the list.
BGP peer	A BGP neighbor. A BGP neighbor is a router running BGP that formed a BGP neighbor relationship to exchange BGP Updates.
BGP peer group	A configuration construct in which multiple neighbors' parameters can be configured as a group to reduce the length of the configuration. BGP performs routing policy logic against one set of Updates for the peer group, improving convergence time and reducing the CPU load.
BGP synchronization	A feature in which internally learned BGP routes cannot be considered to be a best route to reach a prefix unless that same prefix exists in the router's IP routing table as learned via another IGP, such as OSPF or RIP. When synchronization is enabled, BGP does not advertise a route learned via an iBGP peer to external peers, until the route is learned locally, or via an IGP and is present in the routing table. When synchronization is disabled, BGP can use and advertise routes from an iBGP peer to external peers, which are not present in the routing table.
BGP table	A table inside a router that holds the path attributes known by BGP on that router.
Border Gateway Protocol (BGP)	An exterior routing protocol designed to exchange prefix information between different autonomous systems. The information includes characteristics called path attributes, which allow for great flexibility regarding routing choices.

TERM	DESCRIPTION
CIDR	Classless Inter-Domain Routing. A routing method supported by BGP that is based on route aggregation. CIDR allows router to group routes together to cut down on the amount of routing information carried by routers. Several IP networks appear as a single network to networks outside of those grouped together.
External BGP (eBGP)	Refers to how a router views a BGP peer relationship, where the peer is in another AS. See also Internal BGP (iBGP), where the peer is in the same AS.
Interior Gateway Protocol (IGP)	A routing protocol designed to be used to exchange routing information inside a single autonomous system.
Internal BGP (iBGP)	A characteristic of a BGP neighbor relationship, specifically when the two routers are internal to the same BGP ASN.
Neighbor	Another router with which a router decides to exchange routing information.
Neighbor Type	Either external BGP (eBGP), confederation eBGP, or internal BGP (iBGP). The term refers to a peer connection and whether the peers are in different ASs (eBGP), different confederation systems (confederation eBGP), or in the same AS (iBGP).
Next Hop Field	With a routing update, or routing table entry, the portion of a route that defines the next router to which a packet should be sent to reach the destination subnet. With routing protocols, the Next Hop field may define a router other than the router sending the routing update.
Next-hop self	A BGP configuration setting that tells the local router to change the NEXT_HOP path attribute to refer to its own BGP Update Source when advertising routes to BGP neighbors.
Path control	A general term, that refers to any function that impacts how routers forward packets. These functions include routing protocols and any other feature that impacts the IP routing table, plus any feature that impacts the packet forwarding process.
PA (Path Attribute)	Characteristics or attributes of a path to a network. ■ **Well-known Mandatory** PAs must be recognized and present in BGP UPDATE messages. ■ **Well-known Discretionary** PAs must be recognized, but may or may not appear in BGP UPDATE messages. ■ **Optional Transitive** PAs may be supported and are permitted to pass between AS. ■ **Optional Non-Transitive** PAs may be supported but cannot transit an AS (If received from another AS).
Private AS	BGP ASN whose value is between 64,512 and 65,534 (16 bit) and 4,200,000,000 - 4,294,967,294 (32 bit). These values are not assigned for use on the Internet and can be used for private purposes, typically either within confederations or by ISPs to hide the ASN used by some customers.

TERM	DESCRIPTION
A private ASN	An Autonomous System Number (ASN) that falls inside the Private AS range.
Public ASN	An ASN that fits below the private ASN range, specifically from 1 through 64,495 with a 16-bit ASN. With a 32-bit ASN the range is 131,072 through 4,199,999,999.
Redistribution	The process on a router of taking the routes from the IP routing table, as learned by one routing protocol, and injecting routes for those same subnets into another routing protocol.
Regular expression	A list of interspersed alphanumeric literals and metacharacters used to apply complex matching logic to alphanumeric strings. Often used for matching AS_Paths.
Route map	Enables logic to be applied to a set of items. Often used for decisions about what routes to redistribute and for setting characteristics of those routes, for instance, metric values.
Route redistribution	The process of taking routes known through one routing protocol and advertising those routes with another routing protocol.
Route summarization	A consolidation of advertised addresses that causes a single summary route to be advertised.
Summary route **Aggregate route**	A route that is created to represent one or more smaller component routes, typically to reduce the size of routing and topology tables.

BGP path attributes

An important part of the BGP protocol operation is the set of attributes associated with prefixes. Each BGP update message contains a set of attributes. These attributes describe some of the properties of the routes, and can be used in making decisions about which routes to accept and which to reject.

ATTRIBUTE	DESCRIPTION
ORIGIN	How the prefix came to be routed by BGP at the origin AS. The device can learn prefixes from various sources and then put them into BGP. Sources include directly connected interfaces, manually configured static routes, and dynamic internal or external routing protocols. Values are IGP (Interior Gateway Protocol or local networks interior to the originating AS), EGP (other EGPs) and INCOMPLETE (redistributed into BGP or learned by other means). Every update message has this attribute.
AS_PATH	A list of the autonomous systems through which the announcement for the prefix has passed. As prefixes pass between Autonomous Systems, each adds its Autonomous System Number (ASN) to the beginning of the list. This means the AS_path can be used to make routing decisions. Every update message has this attribute, although it may be empty.
NEXT_HOP	The address of the next node to which the device should send packets to get the packets closer to the destination. Every update message has this attribute.

ATTRIBUTE	DESCRIPTION
MULTI-EXIT-DISC	Multi Exit Discriminator. An arbitrary metric assigned to a route, expressing the optimal path by which to reach a particular prefix in or behind a particular AS. One AS sets the value and a different AS uses that value when deciding which path to choose. The MED is also known as the external metric of a route. A lower MED value is preferred over a higher value.
LOCAL_PREF	A metric used in iBGP so each host knows which path inside the AS it should use to reach the advertised prefix. eBGP peers do not send this value, and ignore it on receipt.
ATOMIC_AGGREGATOR	An attribute that allows BGP peers to inform each other about decisions they have made about overlapping routes. If Device A receives overlapping routes, and selects the less specific (more general route) only, then it attaches the atomic_aggregate attribute. When one of its neighbours receives a prefix with the atomic_aggregate attribute set, that neighbour must not take the prefix and de-aggregate it into any more specific entries in BGP.
AGGREGATOR	An attribute that can be attached to an aggregated prefix to specify the AS and Router ID of the device that performed the aggregation.
COMMUNITY	Where the prefix is relevant to and should be advertised to. By default, all prefixes belong to the Internet community, which is the community of all BGP peers. Other communities have been globally defined that limit the scope of prefix advertisement or export, or you can identify a community by a community number.
ORIGINATOR_ID	The Router ID of the iBGP peer that first learned this route, either via an eBGP peer or by some other means such as importing it. This attribute is used by route reflection to prevent routing loops. eBGP peers do not send this value, and ignore it on receipt.
CLUSTER_LIST	A list of the cluster IDs of route reflectors who have reflected the corresponding route(s) within this AS. This attribute is used by route reflection to prevent routing loops. eBGP peers do not send this value, and ignore it on receipt.

The BGP best path algorithm and invalid paths

BGP devices often have more than one path to the same destination. The best path algorithm compares the first **valid** path with the next valid path in the list and decides which is the best route to go into the IP routing table.

The following list outlines the conditions where a path is **not** a valid candidate:

- A Speakers own ASN appears in the AS_PATH advertised from an external BGP peer. (The command "neighbor allowas-in" allows a speaker's own ASN to appear in the AS_PATH.)
- The Next-Hop is unreachable. A Path cannot be selected as best path if the Next-Hop cannot be resolved.
- BGP Synchronization is enabled and no matching prefix exists in the routing table for an internal BGP path. (Path is "not synchronized".)
- Paths which appear in the "show bgp ipv6 neighbors received-routes" command output and are not installed in the BGP table. (The result of neighbor soft-reconfiguration being enabled and the route is denied by inbound BGP policy.)
- Paths which have been dampened (to suppress flapping routes).

Once the path is known to be valid, it steps through the best path algorithm as described in the following table:

STEP	ATTRIBUTE *(in Caps)* *or* FACTOR	VALUE
1	WEIGHT	Highest value. Note: WEIGHT is not a standard RFC Path Attribute. The WEIGHT attribute is local to the device only and is not advertised to peers.
2	LOCAL_PREF	Highest value.
3	Locally Injected Route	Prefer locally originated/injected routes. Preference of locally originated paths: 1. redistribute command 2. network command 3. aggregated-address command
4	AS_PATH	Shortest path.
5	ORIGIN	Lowest (i < e < ?) Preference: 1. (i)IGP is lower than EGP (Exterior Gateway Protocol) 2. (e)EGP is lower than INCOMPLETE 3. (?)INCOMPLETE
6	MULTI_EXIT_DISC	Lowest value. Note: These commands change how MED is evaluated. ■ bgp deterministic-med ■ bgp always-compare-med ■ bgp bestpath med missing-as-worst ■ bgp bestpath med-confed
7	BGP Neighbor Type	Prefer paths learned from eBGP neighbors over iBGP neighbors.
8	IGP Metric to Next-hop	Lowest IGP metric to Next-hop. Note: Paths are considered as BGP equal cost paths if steps 9-12 are used to break the tie.
9	Oldest	First learned / longest known path.
10	Router ID	Lowest neighbor router ID.
11	CLUSTER_LIST	Shortest cluster list length.
12	Peer Address	Lowest peer IPv4/IPv6 address.

BGP Multipath

BGP paths are considered as equal cost paths, if they require steps 9 to12 of the BGP best path algorithm to break the tie. BGP equal cost path candidates must have an identical far left ASN in the AS_PATH. (Both paths must be received from the same neighboring AS.)

The number of equal cost paths installed in the routing table, is determined based on the number of equal cost paths available and the number permitted to be selected. BGP equal cost path candidates are marked as "multipath-candidate" and equal cost path candidates which are installed to the Routing Information Base are marked as "installed".

Note: Even though BGP paths are considered (and can be used) as equal cost paths after step 8, only one becomes the best path as steps 9-12 are used to break the tie.

Example: Viewing a route with multiple equal cost paths:

- In this example we can see that there are three equal paths determined by BGP.
- Only two of these are installed as the **max-paths ebgp 2** command has been set to only allow two paths to be installed.
- Out of these paths, only one is still considered "best".

```
SW1#show bgp ipv6 2001:db8:100::/64
BGP routing table entry for 2001:db8:100::/64
Paths: (3 available, best #3, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
  2001:db8:20::2 2001:db8:30::2
  200
    2001:db8:20::2(fe80::eecd:6dff:fe20:c26b) from 2001:db8:20::2 (2.2.2.2)
    (fe80::eecd:6dff:fe20:c26b)
      Origin IGP metric 0, localpref 100, valid, external, multipath-candidate
      Last update: Mon Jan 13 20:38:15 2014
 200
  2001:db8:30::2(fe80::eecd:6dff:fe20:c26b) from 2001:db8:30::2 (2.2.2.2)
  (fe80::eecd:6dff:fe20:c26b)
    Origin IGP metric 0, localpref 100, valid, external, multipath-candidate, installed
    Last update: Mon Jan 13 20:38:12 2014
200
  2001:db8:10::2(fe80::eecd:6dff:fe20:c26b) from 2001:db8:10::2 (2.2.2.2)
  (fe80::eecd:6dff:fe20:c26b)
    Origin IGP metric 0, localpref 100, valid, external, multipath-candidate, installed, best
    Last update: Mon Jan 13 20:38:09 2014
```

Note: This simulation setup consisted of two AlliedWare Plus Layer 3 switches and three BGP peering sessions to simulate BGP equal cost paths.

This chapter covers the following topics:

- BGP sessions between two devices
- Disabling exchange of IPv4 prefixes over an IPv6 peering
- Enabling IPv6 information exchange to an IPv6 peering
- Establishing a BGP neighborship using the IPv6 link-local address
- Establishing a BGP neighborship using VLAN interfaces
- Setting up BGP neighbors using loopback interfaces as the update source
- Advanced setup options for BGP sessions
- Verification of BGP settings

CHAPTER 2

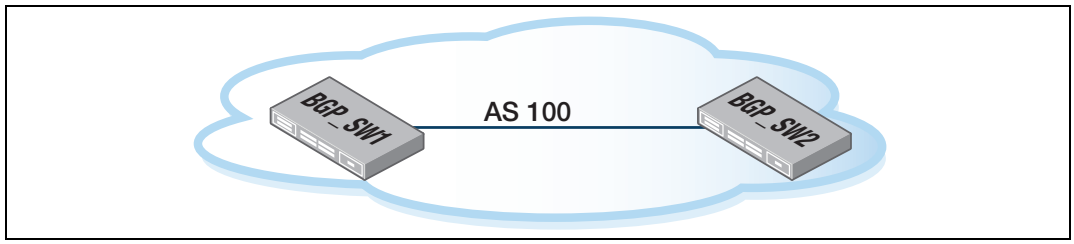
BGP4+ Basic Setup

BGP sessions between two devices

Peering types

- Internal peering (iBGP)
- External peering (eBGP)

Here is an example of iBGP peering.




```
SW1(config)#router bgp 100
SW1(config-router)#no bgp default ipv4-unicast
SW1(config-router)#neighbor 2001:db8:10::2 remote-as 100

SW2(config)#router bgp 100
SW2(config-router)#no bgp default ipv4-unicast
SW2(config-router)#neighbor 2001:db8:10::1 remote-as 100
```

Here is an example of eBGP peering.


```
SW1(config)#router bgp 100
SW1(config-router)#no bgp default ipv4-unicast
SW1(config-router)#neighbor 2001:db8:10::2 remote-as 200

SW2(config)#router bgp 200
SW2(config-router)#no bgp default ipv4-unicast
SW2(config-router)#neighbor 2001:db8:10::1 remote-as 100
```

Peering methods

Depending on the desired solution, the following solutions can be used for the peering:

- Loopback interface global IPv6 address.
- VLAN interface global IPv6 address.
- VLAN interface link-local IPv6 address.

Which BGP peering method to use?

Looking at the available peering methods available, network engineers should take care to select which peering method would be the best to use for each situation.

When to use IPv6 Link Local Address for peering:

- When an eBGP peer is directly connected on the link.
- When conserving IPv6 address space and IPv6 hardware table entries.

Here is an example using each BGP devices VLAN interface link-local address for BGP peering.

```
SW1(config)#router bgp 100
SW1(config-router)#neighbor fe80::2 remote-as 200
SW1(config-router)#neighbor fe80::2 interface vlan10

SW2(config)#router bgp 200
SW2(config-router)#neighbor fe80::1 remote-as 100
SW2(config-router)#neighbor fe80::1 interface vlan10
```

Advantages

- IPv6 address space and entries are conserved.
- Public IPv6 addresses are not wasted for BGP sessions on private links.

Considerations

- Peers must be directly connected on the link.
- May not be suitable for public links.
- Exit interface must be defined for Link Local peering.

When to use VLAN interface with IPv6 Global Unicast Address for BGP peering:

- When an eBGP peering is conducted that may span several hops.
- When a single interface is used for the peering only.

Here is an example using each BGP devices VLAN interface global unicast address for eBGP peering.

```
SW1(config)#router bgp 100
SW1(config-router)#no bgp default ipv4-unicast
SW1(config-router)#neighbor 2001:db8:10::2 remote-as 200

SW2(config)#router bgp 200
SW2(config-router)#no bgp default ipv4-unicast
SW2(config-router)#neighbor 2001:db8:10::1 remote-as 100
```

Advantages

- Peering may be achieved to an eBGP peer, which is more than one hop away.
- This peering method is suitable for both iBGP and eBGP sessions as well as public and private links.

Considerations

- Redundancy must be achieved via another layer using link aggregation, as a VLAN interface does not have the same level of resiliency as a loopback interface.
- If multiple paths exist to a peering device, it is inefficient to setup several BGP sessions to the neighboring device to achieve redundancy. This results in multiple sessions being maintained as well as routes advertised and received on each session. In this circumstance, peer the devices using the Loopback interfaces.

When to use Loopback interface with IPv6 Global Unicast Address for BGP peering:

- This peering method builds on top of the capabilities offered by peering with Layer 3 VLAN interfaces.
- When there are multiple equal cost routes used by the BGP session to reach a neighbor's loopback interface.

Advantages

- This peering method introduces redundancy and resiliency as the loopback interface never goes down. Using a loopback interface which is reachable via redundant links, allows for remote peers to re-route BGP messages to the local speaker without having to tear the session down.
- When redundant equal cost routes to reach the neighbor's loopback interface exist, the BGP session (and traffic) can be load balanced across both links recursively.
- This peering method is suitable for both iBGP and eBGP sessions as well as public and private links.

Considerations

- Deployment in iBGP may be advantageous due to a more meshed environment.
- An IGP route to the peer loopback subnet must exist to allow peering session between loopback interfaces to establish.

Deployment in eBGP may be advantageous when redundant paths or equal cost links are available.

Here is an example using each BGP devices loopback interface global unicast address for eBGP peering.

```
SW1(config)#router bgp 100
SW1(config-router)#no bgp default ipv4-unicast
SW1(config-router)#neighbor 2001:db8:2::1 remote-as 200
SW1(config-router)#neighbor 2001:db8:2::1 update-source lo
SW1(config-router)#neighbor 2001:db8:2::1 ebgp-multihop 2

SW2(config)#router bgp 200
SW2(config-router)#no bgp default ipv4-unicast
SW2(config-router)#neighbor 2001:db8:1::1 remote-as 100
SW2(config-router)#neighbor 2001:db8:1::1 update-source lo
SW2(config-router)#neighbor 2001:db8:1::1 ebgp-multihop 2
```

Disabling exchange of IPv4 prefixes over an IPv6 peering

Neighbors and peer groups defined under the BGP routing process are considered as IPv4 unicast neighbors and will exchange IPv4 prefixes via the IPv6 the peering by default.

To prevent this happening, two solutions are available:

Solution #1: Disable exchange of IPv4 information per ipV6 peer or ipV6 peer group.

In the example below, exchange of IPv4 information will be disabled to the IPv6 neighbor 2001:db8:10::2.

IPv4 information will still be exchanged with the configured IPv4 neighbor 192.168.10.2.

Example:

```
SW1(config)#router bgp 65536
SW1(config-router)#no neighbor 2001:db8:10::2 activate

!
SW1#show run bgp
!
router bgp 65536
bgp router-id 1.1.1.1
network 172.16.0.0/16
neighbor 192.168.10.2 remote-as 65537
neighbor 2001:db8:10::2 remote-as 65537
no neighbor 2001:db8:10::2 activate
!
address-family ipv6
network 2001:db8:100::/64
neighbor 2001:db8:10::2 activate
exit-address-family
!
```

Solution #2: Disable exchange of IPv4 information by default.

This solution changes the default exchange of IPv4 information to neighbors or peer groups, from enabled to disabled. Each neighbor or peer group configured for which exchange of IPv4 information is desired, must be manually activated in the BGP router configuration mode.

Note: This must be configured first before IPv4 and IPv6 neighbors are defined. The command will only disable IPv4 information exchange for neighbors configured after this command has been applied.

If the command **no bgp default ipv6-unicast** is applied after peers have been configured, Allied Ware Plus assumes that existing peers are already permitted for the exchange of IPv4 prefixes. You must then manually disable IPv4 information exchange per preconfigured IPv6 peer. (As seen in Solution #1.)

Hidden configuration

Some commands in the running configuration which enable certain behaviors are hidden by default.

- When **bgp default-ipv4-unicast** is **enabled** (default), the "neighbor <neighbor addr / peer group> activate" is hidden.
- When **bgp default-ipv4-unicast** is **disabled**, the "no neighbor <peer address> activate" is hidden.

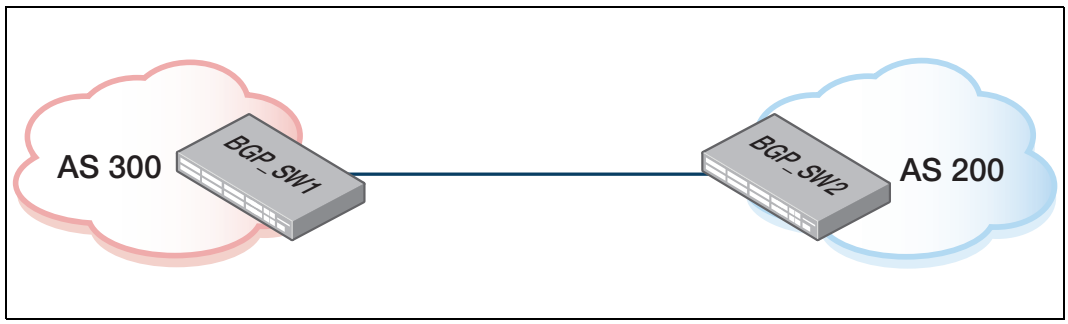
For example, when **bgp default-ipv4-unicast** is disabled, the "no neighbor 192.168.10.2 activate" command is hidden in the running configuration, as it is now the default.

```
SW1#show run bgp
!
router bgp 65536
bgp router-id 1.1.1.1
no bgp default ipv4-unicast
neighbor 192.168.10.2 remote-as 65537
neighbor 2001:db8:10::2 remote-as 65537
!
address-family ipv6
neighbor 2001:db8:10::2 activate
exit-address-family
!
```

We must manually enable the IPv4 unicast information exchange to the IPv4 peer.

```
SW1(config)#router bgp 65536
SW1(config-router)#neighbor 192.168.10.2 activate

SW1#show run bgp
!
router bgp 65536
bgp router-id 1.1.1.1
no bgp default ipv4-unicast
network 172.16.0.0/16
neighbor 192.168.10.2 remote-as 65537
neighbor 192.168.10.2 activate
neighbor 2001:db8:10::2 remote-as 65537
!
address-family ipv6
network 2001:db8:100::/64
neighbor 2001:db8:10::2 activate
exit-address-family
!
```

Enabling IPv6 information exchange to an IPv6 peer



Use

The neighbor activate command enables information (routes) exchange to a specified peer in the corresponding address family.

Objective

- Activate BGP4+ information exchange to a peer (attributes, communities, end of RIB marker etc.).
- This simulation assumes that there is already prefixes being advertised into BGP.

1. Specify a BGP multiprotocol neighbor.

```
BGP_SW1(config)#router bgp 300
BGP_SW1(config-router)#no bgp default ipv4-unicast
BGP_SW1(config-router)#neighbor 2001:db8:b::1 remote-as 200
```

2. Activate BGP4+ information exchange to the peer.

```
BGP_SW1(config-router)#address-family ipv6
BGP_SW1(config-router-af)#neighbor 2001:db8:b::1 activate
```

Verify by viewing the:

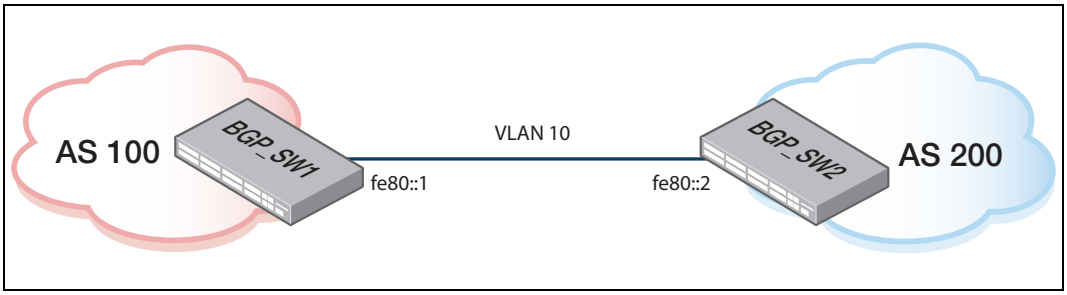
- BGP neighbor connections (check PfxRcd counter).
- BGP table (check prefixes from peer have populated the table).
- BGP neighbor information (check accepted and announced prefix counters).

```
BGP_SW1#show bgp ipv6 summary
BGP router identifier 3.3.3.1, local AS number 300
BGP table version is 1247
3 BGP AS-PATH entries
0 BGP community entries

Neighbor        V         AS   MsgRc  MsgSnt TblVer InOutQ Up/Down State/PfxRcd
2001:db8:b::1   4        200    2952    2946   1246 0/0    01:13:42       18

Number of neighbors 1
```

```
BGP_SW1#show bgp ipv6
BGP table version is 1249, local router ID is 3.3.3.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> ::/0             2001:db8:b::1            0             0 200 100 i
*> 2001:db8:1::/64  2001:db8:b::1            0             0 200 ?
*> 2001:db8:2::/64  2001:db8:b::1            0             0 200 ?
*  2001:db8:3::/64  2001:db8:b::1            0             0 200 ?
*>                  ::                                   32768 ?
*  2001:db8:4::/64  2001:db8:b::1            0             0 200 ?
*>                  ::                                   32768 ?
*> 2001:db8:6::/64  2001:db8:b::1            0             0 200 100 i
*> 2001:db8:a:10::/64
                    2001:db8:b::1            0             0 200 100 i
*> 2001:db8:a:20::/64
                    2001:db8:b::1            0             0 200 100 i
*> 2001:db8:a:30::/64
                    2001:db8:b::1            0             0 200 100 i
*> 2001:db8:a:40::/64
                    2001:db8:b::1            0             0 200 100 i
*> 2001:db8:a:50::/64
                    2001:db8:b::1            0             0 200 100 i
*> 2001:db8:a:60::/64
                    2001:db8:b::1            0             0 200 100 i
*> 2001:db8:a:70::/64
                    2001:db8:b::1            0             0 200 100 i
*> 2001:db8:b::1/128
                    2001:db8:b::1            0             0 200 ?
*> 2001:db8:b:10::/64
                    2001:db8:b::1            0             0 200 ?
*> 2001:db8:b:20::/64
                    2001:db8:b::1            0             0 200 ?
*> 2001:db8:b:30::/64
                    2001:db8:b::1            2             0 200 ?
*> 2001:db8:b:40::/64
                    2001:db8:b::1            2             0 200 ?
*> 2001:db8:c::1/128
*                   ::                                      32768 ?
*> 2001:db8:c:10::/64
                    ::                                      32768 ?
*> 2001:db8:c:20::/64
                    ::                                      32768 ?
*> 2001:db8:c:30::/64
                    ::                                  100 32768 i
*> 2001:db8:c:40::/64
                   ::                                   100 32768 i
Number of prefixes 23
```

```
BGP_SW1#show bgp ipv6 neighbors 2001:db8:b::1
BGP neighbor is 2001:db8:b::1, remote AS 200, local AS 300, external link
  BGP version 4, remote router ID 2.2.2.1
  BGP state = Established, up for 01:16:50
  Last read 01:16:50, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    4-Octet ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 2952 messages, 8 notifications, 0 in queue
  Sent 2939 messages, 14 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
  Update source is lo
For address family: IPv6 Unicast
  BGP table version 1250, neighbor version 1249
  Index 2, Offset 0, Mask 0x4
  AF-dependant capabilities:
  Community attribute sent to this neighbor (both)
  18 accepted prefixes
  7 announced prefixes

 Connections established 15; dropped 14
  External BGP neighbor may be up to 2 hops away.
Local host: 2001:db8:c::1, Local port: 51710
Foreign host: 2001:db8:b::1, Foreign port: 179
Nexthop: 3.3.3.1
Nexthop global: 2001:db8:c::1
Nexthop local: ::
BGP connection: non shared network
Last Reset: 01:16:50, due to BGP Notification received
Notification Error Message: (OPEN Message Error/Unsupported Optional Parameter.)
```

Establishing a BGP neighborship using the IPv6 link-local address



Objective

Form a BGP neighborship between two AlliedWare Plus devices.

1. Configure the BGP Router-ID.

```
SW1(config)#router bgp 100
SW1(config-router)#bgp router-id 1.1.1.1

SW2(config)#router bgp 200
SW2(config-router)#bgp router-id 2.2.2.2
```

2. Define the BGP neighbor.

```
SW1(config-router)#neighbor fe80::2 remote-as 200
SW2(config-router)#neighbor fe80::1 remote-as 100
```

3. Set the interface used to reach the link-local BGP neighbor.

```
SW1(config-router)#neighbor fe80::2 interface vlan10
SW2(config-router)#neighbor fe80::1 interface vlan10
```

4. Enable the exchange of information to each BGP neighbor.

```
SW1(config-router)#address-family ipv6
SW1(config-router-af)#neighbor fe80::2 activate

SW2(config-router)#address-family ipv6
SW2(config-router-af)#neighbor fe80::1 activate
```

Verify

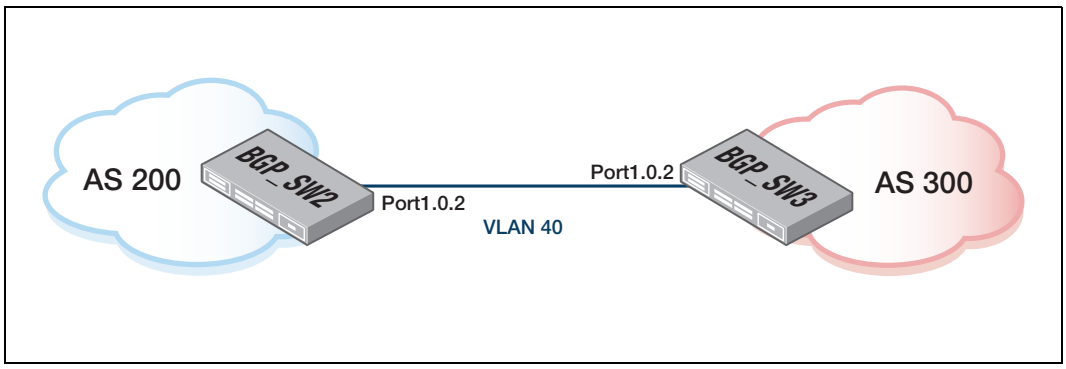
- Verify by viewing the BGP neighbours on each peer.
- Notice the next hop global address is the link local address of the peer.

```
SW1#show bgp ipv6 neighbors
BGP neighbor is fe80::2, remote AS 200, local AS 100, external link
 BGP version 4, remote router ID 2.2.2.2
 BGP state = Established, up for 00:00:50
 Last read 00:00:50, hold time is 90, keepalive interval is 30 seconds
 Neighbor capabilities:
   Route refresh: advertised and received (old and new)
   4-Octet ASN Capability: advertised and received
   Address family IPv4 Unicast: advertised and received
   Address family IPv6 Unicast: advertised and received
 Received 7 messages, 1 notifications, 0 in queue
 Sent 7 messages, 0 notifications, 0 in queue
 Route refresh request: received 0, sent 0
 Minimum time between advertisement runs is 30 seconds
For address family: IPv4 Unicast
 BGP table version 1, neighbor version 1
 Index 1, Offset 0, Mask 0x2
 Community attribute sent to this neighbor (both)
 0 accepted prefixes
 0 announced prefixes
For address family: IPv6 Unicast
 BGP table version 1, neighbor version 1
 Index 1, Offset 0, Mask 0x2
 Community attribute sent to this neighbor (both)
 0 accepted prefixes
 0 announced prefixes

Connections established 2; dropped 1
Local host: fe80::1, Local port: 179
Foreign host: fe80::2, Foreign port: 34459
Nexthop: 1.1.1.1
Nexthop global: fe80::1
Nexthop local: ::
BGP connection: shared network
Last Reset: 00:00:50, due to BGP Notification received
Notification Error Message: (Cease/Other Configuration Change.)
```

```
SW2#show bgp ipv6 neighbors
BGP neighbor is fe80::1, remote AS 100, local AS 200, external link
 BGP version 4, remote router ID 1.1.1.1
 BGP state = Established, up for 00:02:12
 Last read 00:02:12, hold time is 90, keepalive interval is 30 seconds
 Neighbor capabilities:
  Route refresh: advertised and received (old and new)
   4-Octet ASN Capability: advertised and received
   Address family IPv4 Unicast: advertised and received
   Address family IPv6 Unicast: advertised and received
 Received 11 messages, 0 notifications, 0 in queue
 Sent 10 messages, 1 notifications, 0 in queue
 Route refresh request: received 0, sent 0
 Minimum time between advertisement runs is 30 seconds
For address family: IPv4 Unicast
 BGP table version 1, neighbor version 1
 Index 1, Offset 0, Mask 0x2
 Community attribute sent to this neighbor (both)
 0 accepted prefixes
 0 announced prefixes
For address family: IPv6 Unicast
 BGP table version 1, neighbor version 1
 Index 1, Offset 0, Mask 0x2
 Community attribute sent to this neighbor (both)
 0 accepted prefixes
 0 announced prefixes

Connections established 2; dropped 1
Local host: fe80::2, Local port: 34459
Foreign host: fe80::1, Foreign port: 179
Nexthop: 2.2.2.2
Nexthop global: fe80::2
Nexthop local: ::
BGP connection: shared network
Last Reset: 00:02:12, due to BGP Notification sent
Notification Error Message: (Cease/Other Configuration Change.)
```

Establish a BGP neighborship using VLAN interfaces

Scenario



Objective

Form a BGP neighborship between two AlliedWare Plus switches.

1. Create a VLAN to use for the BGP neighborship.

```
BGP_SW2(config)#vlan database
BGP_SW2(config-vlan)#vlan 40
BGP_SW2(config-vlan)#exit

BGP_SW3(config)#vlan database
BGP_SW3(config-vlan)#vlan 40
BGP_SW3(config-vlan)#exit
```

2. Create VLAN 40 on each switch and apply IPv6 addressing.

```
BGP_SW2(config)#interface vlan40
BGP_SW2(config-if)#ipv6 address 2001:db8:40::1/64

BGP_SW2(config)#interface lo
BGP_SW2(config-if)#ipv6 address 2001:db8:2::1/64

BGP_SW3(config)#interface vlan40
BGP_SW3(config-if)#ipv6 address 2001:db8:40::2/64
```

3. Associate the VLAN with an access port or trunk.

Use the switchport in access mode, if only this VLAN is required on the link.

```
BGP_SW2(config)#interface port1.0.2
BGP_SW2(config-if)#switchport
BGP_SW2(config-if)#switchport mode access
BGP_SW2(config-if)#switchport access vlan 40

BGP_SW3(config)#interface port1.0.2
BGP_SW3(config-if)#switchport
BGP_SW3(config-if)#switchport mode access
BGP_SW3(config-if)#switchport access vlan 40
```

Use the 802.1Q trunk if additional VLANs will be required on the link.

```
BGP_SW2(config)#interface port1.0.2
BGP_SW2(config-if)#switchport mode trunk
BGP_SW2(config-if)#switchport trunk allowed vlan add 40
BGP_SW2(config-if)#switchport trunk native vlan none

BGP_SW3(config)#interface port1.0.2
BGP_SW3(config-if)#switchport mode trunk
BGP_SW3(config-if)#switchport trunk allowed vlan add 40
BGP_SW3(config-if)#switchport trunk native vlan none
```

4. Setup BGP processing.

- Configure a BGP router ID and assign the neighbor as being in an external AS.
- Under the IPv6 address family, activate the neighbor to enable the exchange of information.

```
BGP_SW2(config)#router bgp 200
BGP_SW2(config-router)#bgp router-id 2.2.2.2
BGP_SW2(config-router)#no bgp default ipv4-unicast
BGP_SW2(config-router)#neighbor 2001:db8:40::2 remote-as 300
BGP_SW2(config-router)#address-family ipv6
BGP_SW2(config-router-af)#neighbor 2001:db8:40::2 activate
```

```
BGP_SW3(config) #router bgp 300
BGP_SW3(config-router)#bgp router-id 3.3.3.3
BGP_SW3(config-router)#no bgp default ipv4-unicast
BGP_SW3(config-router)#neighbor 2001:db8:40::1 remote-as 200
BGP_SW3(config-router)#address-family ipv6
BGP_SW3(config-router-af)#neighbor 2001:db8:40::1 activate
```

Additional hops

The IPv6 forwarding (unicast routing) command must be configured if additional hops are added to the network.

Verify using the commands:

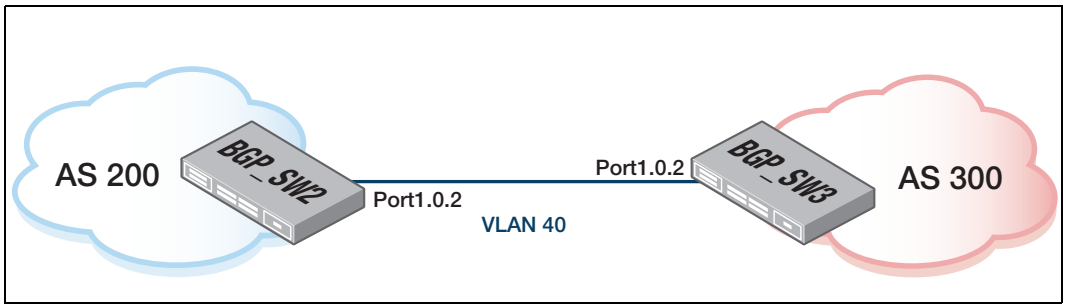
- show bgp ipv6 neighbors
- show bgp ipv6 summary

```
BGP_SW2#show bgp ipv6 neighbors
BGP neighbor is 2001:db8:40::2, remote AS 300, local AS 200, external link
  BGP version 4, remote router ID 3.3.3.3
  BGP state = Established, up for 00:00:07
  Last read 00:00:07, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    4-Octet ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 2 messages, 0 notifications, 0 in queue
  Sent 2 messages, 0 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
 For address family: IPv6 Unicast
  BGP table version 1, neighbor version 1
  Index 0, Offset 0, Mask 0x1
  Community attribute sent to this neighbor (both)
  0 accepted prefixes
  0 announced prefixes

 Connections established 1; dropped 0
Local host: 2001:db8:40::1, Local port: 55479
Foreign host: 2001:db8:40::2, Foreign port: 179
Nexthop: 2.2.2.2
Nexthop global: 2001:db8:40::1
Nexthop local: fe80::209:41ff:fefb:c323
BGP connection: shared network
```

```
BGP_SW2#show bgp ipv6 summary
BGP router identifier 2.2.2.2, local AS number 200
BGP table version is 1
0 BGP AS-PATH entries
0 BGP community entries

Neighbor        V         AS   MsgRc  MsgSnt TblVer InOutQ Up/Down State/PfxRcd
2001:db8:40::2  4         300       4       4       1 0/0    00:01:06        0

Number of neighbors 1
```

Setting up BGP neighbors using loopback interfaces as the update source

Scenario



Objective

Form a BGP adjacency between two AlliedWare Plus switches.

1. Create a VLAN to use for the BGP session to transit.

```
BGP_SW2(config)#vlan database
BGP_SW2(config-vlan)#vlan 40
BGP_SW2(config-vlan)#exit

BGP_SW3(config)#vlan database
BGP_SW3(config-vlan)#vlan 40
BGP_SW3(config-vlan)#exit
```

2. Create VLAN 40, apply IPv6 addressing to VLAN 40 and loopback in each switch.

```
BGP_SW2(config)#interface vlan40
BGP_SW2(config-if)#ipv6 address 2001:db8:40::1/64

BGP_SW2(config)#interface lo
BGP_SW2(config-if)#ipv6 address 2001:db8:2::1/64

BGP_SW3(config)#interface vlan40
BGP_SW3(config-if)#ipv6 address 2001:db8:40::2/64

BGP_SW3(config)#interface lo
BGP_SW3(config-if)#ipv6 address 2001:db8:3::1/64
```

3. Associate the VLAN with an access port or trunk.

Use the switchport in access mode if only this VLAN is required on the link.

```
BGP_SW2(config)#interface port1.0.2
BGP_SW2(config-if)#switchport
BGP_SW2(config-if)#switchport mode access
BGP_SW2(config-if)#switchport access vlan 40

BGP_SW3(config)#interface port1.0.2
BGP_SW3(config-if)#switchport
BGP_SW3(config-if)#switchport mode access
BGP_SW3(config-if)#switchport access vlan 40
```

Use the 802.1Q trunk if additional VLANs will be required on the link.

```
BGP_SW2(config)#interface port1.0.2
BGP_SW2(config-if)#switchport mode trunk
BGP_SW2(config-if)#switchport trunk allowed vlan add 40
BGP_SW2(config-if)#switchport trunk native vlan none

BGP_SW3(config)#interface port1.0.2
BGP_SW3(config-if)#switchport mode trunk
BGP_SW3(config-if)#switchport trunk allowed vlan add 40
BGP_SW3(config-if)#switchport trunk native vlan none
```

4. Setup static routes to enable loopback interfaces to be reachable on each switch.

- The best practise is to use the link-local address as the Next Hop because it will not change.
- Because the link-local address is used as the Next Hop, the exit interface must also be specified.

```
BGP_SW2(config)#ipv6 route 2001:db8:3::1/128 fe80::200:cdff:fe2a:e635
vlan40
BGP_SW3(config)#ipv6 route 2001:db8:2::1/128 fe80::209:41ff:fefb:c323
vlan40
```

5. Setup BGP.

- Configure a BGP router ID and assign the neighbor as being in an external AS.
- eBGP multihop must be used if the neighbor external to the AS is not directly connected.
- The loopback is to be set as the update source for BGP.
- Under the IPv6 address family, activate the neighbor to enable the exchange of information.

```
BGP_SW2(config)#router bgp 200
BGP_SW2(config-router)#bgp router-id 2.2.2.2
BGP_SW2(config-router)#no bgp default ipv4-unicast
BGP_SW2(config-router)#neighbor 2001:db8:3::1 remote-as 300
BGP_SW2(config-router)#neighbor 2001:db8:3::1 ebgp-multihop 2
BGP_SW2(config-router)#neighbor 2001:db8:3::1 update-source lo
BGP_SW2(config-router)#address-family ipv6
BGP_SW2(config-router-af)#neighbor 2001:db8:3::2 activate

BGP_SW3(config)#router bgp 300
BGP_SW3(config-router)#bgp router-id 3.3.3.3
BGP_SW1(config-router)#no bgp default ipv4-unicast
BGP_SW2(config-router)#neighbor 2001:db8:2::1 remote-as 200
BGP_SW2(config-router)#neighbor 2001:db8:2::1 ebgp-multihop 2
BGP_SW2(config-router)#neighbor 2001:db8:2::1 update-source lo
BGP_SW3(config-router)#address-family ipv6
BGP_SW3(config-router-af)#neighbor 2001:db8:2::1 activate
```

Additional hops

IPv6 forwarding (unicast routing) must be configured if additional hops are added to the network.

Verify using the commands:

- show bgp ipv6 neighbors
- show bgp ipv6 summary

```
BGP_SW2#show bgp ipv6 neighbors 2001:db8:3::1
BGP neighbor is 2001:db8:3::1, remote AS 300, local AS 200, external link
  BGP version 4, remote router ID 3.3.3.3
  BGP state = Established, up for 00:19:39
  Last read 00:19:39, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    4-Octet ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 48 messages, 0 notifications, 0 in queue
  Sent 48 messages, 0 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
  Update source is lo
 For address family: IPv6 Unicast
  BGP table version 1, neighbor version 1
  Index 2, Offset 0, Mask 0x4
  Community attribute sent to this neighbor (both)
  0 accepted prefixes
  0 announced prefixes

 Connections established 1; dropped 0
  External BGP neighbor may be up to 2 hops away.
Local host: 2001:db8:2::1, Local port: 51269
Foreign host: 2001:db8:3::1, Foreign port: 179
Nexthop: 2.2.2.2
Nexthop global: 2001:db8:2::1
Nexthop local: ::
BGP connection: non shared network
```

```
BGP_SW2#show bgp ipv6 summary
BGP router identifier 2.2.2.2, local AS number 200
BGP table version is 1
0 BGP AS-PATH entries
0 BGP community entries

Neighbor        V         AS   MsgRc  MsgSnt TblVer InOutQ Up/Down State/PfxRcd
2001:db8:3::1   4        300      48      48      1 0/0    00:19:53        0

Number of neighbors 1
```

Advanced setup options for BGP sessions

Description

It is advisable to add descriptions to defined neighbors so anyone using the switch can see who the device is BGP peered to/with.

Example

```
SW1(config-router)#neighbor 2001:db8:10::2 description ISP#1
```

Timers

- AlliedWare Plus BGP uses the session times as defined / recommended in the RFC.
- These values can be tuned or changed depending on network requirements.

AlliedWare Plus default timers

All values are in accordance with RFC4271.

```
Keepalive = 30 seconds
Holdtimer = 90 (3*keepalive timer unless manually defined)
Connect = 120
Route advertisement interval (eBGP peers) = 30 seconds
Route advertisement interval (iBGP peers) = 5 seconds
AS origination interval (iBGP peers) = 15 seconds
```

To change keepalive and holdtimers, use the following command:

```
SW1(config-router)#neighbor 2001:db8:10::2 timers 20 60
```

To change connect timer values per neighbor, use the following command:

```
SW1(config-router)#neighbor 2001:db8:10::2 timers connect 60
```

To change the AS origination interval per neighbor, use the following command:

```
SW1(config-router)#neighbor 2001:db8:10::2 as-origination-interval 10
```

To change the route advertisement interval per neighbor, use the following command:

```
SW1(config-router)#neighbor 2001:db8:10::2 advertisement-interval 20
```

BGP port

- The default port used to connect to BGP peers is TCP 179.
- Using AlliedWare Plus you can specify a different destination port for the TCP session.

```
SW1(config-router)#neighbor 2001:db8:10::2 port 20000
```

Security

- By default there is no authentication applied.
- To setup MD5 authentication for the BGP peering session, it must be applied on each peer.

To setup authentication for a peering example, use the following command:

```
SW1(config-router)#neighbor 2001:db8:10::2 password 1234
```

Soft reconfiguration

This older method of clearing routes may be required if the peer doesn't support route refresh. This method creates a copy of the Adj-RIB-In and stores it in memory. Here is a soft reconfiguration example:

```
SW1(config-router)#neighbor 2001:db8:10::2 soft-reconfiguration inbound
```

Verification of BGP settings

Most of the BGP settings can be verified using the show bgp ipv6 neighbors command.

The command can verify information about the:

- Session.
- IPv4 Unicast Address Family.
- IPv6 Unicast Address Family.
- Connection.

Session information

- Neighbor address, ASN information, and if it is an internal or external link.
- BGP version and state.
- Neighbor capabilities for the BGP session.
- Number of messages transmitted and received.

IPv4 Unicast Address Family information

- BGP table version.
- IPv4 AF dependant capabilities.
- Communities.
- Route filters for updates on ingress and egress.
- Number of announced and accepted prefixes.

IPv6 Unicast Address Family information

- BGP table version.
- IPv6 AF dependant capabilities.
- Communities.
- Route filters for updates on ingress and egress.
- Number of announced and accepted prefixes.

Connection information

- Connection counters.
- Graceful restart timer.
- Hop count to the peer.
- Next Hop information.
- Local and external port numbers.

Example output

```
BGP_SW2#show bgp ipv6 neighbors 2001:db8:b::1
BGP neighbor is 2001:db8:b::1, remote AS 200, local AS 100, external link
  BGP version 4, remote router ID 2.2.2.1
  BGP state = Established, up for 01:03:26
  Last read 01:03:26, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
   Route refresh: advertised and received (old and new)
   4-Octet ASN Capability: advertised and received
   Address family IPv6 Unicast: advertised and received
 Received 157 messages, 0 notifications, 0 in queue
 Sent 228 messages, 0 notifications, 0 in queue
 Route refresh request: received 0, sent 0
 Minimum time between advertisement runs is 30 seconds
 Update source is lo
For address family: IPv6 Unicast
 BGP table version 66, neighbor version 66
 Index 2, Offset 0, Mask 0x4
 AF-dependant capabilities:
 Graceful restart: advertised, received
 Community attribute sent to this neighbor (both)
 Default information originate, default sent
 Inbound path policy configured
 Incoming update prefix filter list is *BGP_FILTER_LIST
 Route map for incoming advertisements is *BGP_LOCAL_PREF_MAP
 8 accepted prefixes
 8 announced prefixes

Connections established 1; dropped 0
Graceful-restart Status:
Remote restart-time is 90 sec
 External BGP neighbor may be up to 2 hops away.
Local host: 2001:db8:a::1, Local port: 179
Foreign host: 2001:db8:b::1, Foreign port: 50672
Nexthop: 1.1.1.1
Nexthop global: 2001:db8:a::1
Nexthop local: ::
BGP connection: non shared network
```

- Redistributing all known OSPFv3 routes into BGP
- Redistributing selected OSPFv3 routes into BGP using a route map with the redistribution statement
- Using a route map with BGP default-originate to advertise a default route conditionally
- Using redistribution to dynamically advertise RIPng routes into BGP
- Using a route map to redistribute specific RIPng routes into BGP

This chapter covers the following topics:

- Redistributing all known OSPFv3 routes into BGP
- Redistributing selected OSPFv3 routes into BGP using a route map with the redistribution statement
- Using a route map with BGP default-originate to advertise a default route conditionally
- Using redistribution to dynamically advertise RIPng routes into BGP
- Using a route map to redistribute specific RIPng routes into BGP

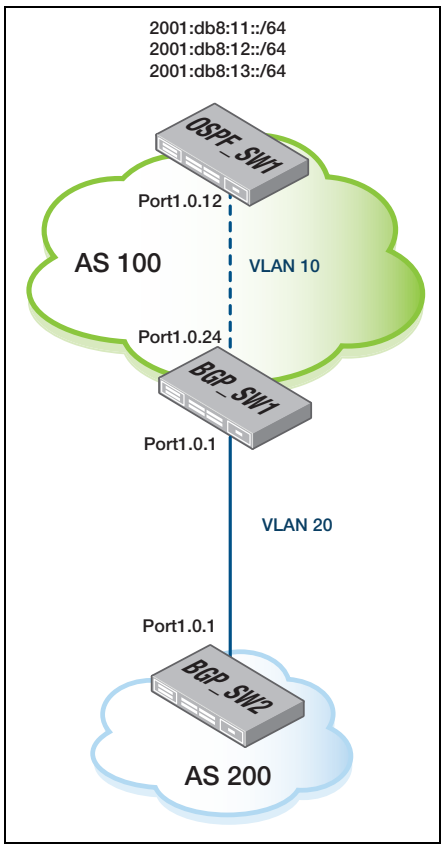
CHAPTER 3

Redistribution

Redistribution is used by routing protocols to advertise routes that are learned by some other means, such as by another routing protocol or by static or connected routes. Since all internal routes are imported into BGP, careful filtering is applied to make sure that only routes to be advertised reach the Internet, not everything. The redistribution command injects prefixes from one routing protocol into another routing protocol.

Redistributing all known OSPFv3 routes into BGP

Scenario



Objective

Redistribute all known External OSPFv3 routes (2001:db8:11::/64, 2001:db8:12::/64, 2001:db8:13::/64) into BGP.

Prerequisite

- OSPFv3 adjacency must be formed between OSPF_SW1 and BGP_SW1.
- BGP_SW1 knows of external networks advertised by OSPFv3 neighbor 2.2.2.2 (OSPF_SW1).

```
BGP_SW1#show running-config router ipv6 ospf
!
router ipv6 ospf
 router-id 1.1.1.1
 no bgp default ipv4-unicast
!


BGP_SW1#show running-config interface vlan10
!
interface vlan10
 ipv6 address 2001:db8:10::1/64
 ipv6 router ospf area 0
!
```

```
BGP_SW1#show ipv6 ospf neighbor
OSPFv3 Process (*null*)
Neighbor ID     Pri   State           Dead Time   Interface  Instance ID
2.2.2.2           1   Full/DR         00:00:33    vlan10     0


BGP_SW1#show ipv6 ospf route
OSPFv3 Process (*null*)
Codes: C - connected, D - Discard, O - OSPF, IA - OSPF inter area
       E1 - OSPF external type 1, E2 - OSPF external type 2

   Destination       Metric Next-hop
C  2001:db8:10::/64  1      directly connected, vlan10, Area 0.0.0.0
E2 2001:db8:11::/64  1/20   via fe80::200:cdff:fe28:7e6, vlan10
E2 2001:db8:12::/64  1/20   via fe80::200:cdff:fe28:7e6, vlan10
E2 2001:db8:13::/64  1/20   via fe80::200:cdff:fe28:7e6, vlan10
```

1. Redistribute all known OSPFv3 networks into BGP.

```
BGP_SW1(config)#router bgp 100
BGP_SW1(config-router)#address-family ipv6
BGP_SW1(config-router-af)#redistribute ospf
```

```
BGP_SW1#show running-config bgp
!
router bgp 100
bgp router-id 1.1.1.1
no bgp default ipv4-unicast
neighbor 2001:db8:2::1 remote-as 200
neighbor 2001:db8:2::1 ebgp-multihop 2
neighbor 2001:db8:2::1 update-source lo
!
address-family ipv6
redistribute ospf
neighbor 2001:db8:2::1 activate
exit-address-family
!
```

Verify by viewing the:

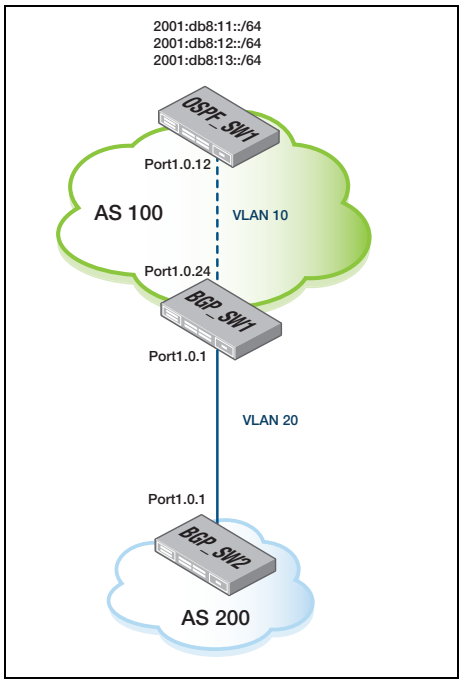
- BGP table on BGP_SW2.
- IPv6 routing table on BGP_SW2.

```
BGP_SW2#show bgp ipv6
BGP table version is 6, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:11::/64 2001:db8:1::1           20             0 100 ?
*> 2001:db8:12::/64 2001:db8:1::1           20             0 100 ?
*> 2001:db8:13::/64 2001:db8:1::1           20             0 100 ?
Number of prefixes 3


BGP_SW2#show ipv6 route
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
       IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
Timers: Uptime

S     2001:db8:1::1/128 [1/0] via fe80::200:cdff:fe2a:eaf1, vlan20, 00:41:45
C     2001:db8:2::/64 via ::, lo, 00:41:45
B    2001:db8:11::/64 [20/20] via 2001:db8:1::1 (recursive via fe80::200:cdff:fe2a:eaf1), vlan20,
00:07:35
B    2001:db8:12::/64 [20/20] via 2001:db8:1::1 (recursive via fe80::200:cdff:fe2a:eaf1), vlan20,
00:07:35
B    2001:db8:13::/64 [20/20] via 2001:db8:1::1 (recursive via fe80::200:cdff:fe2a:eaf1), vlan20,
00:07:35
C     2001:db8:20::/64 via ::, vlan20, 00:41:37
C     fe80::/64 via ::, vlan20, 00:41:37
C     fe80::/64 via ::, lo, 00:41:45
```

Redistributing selected OSPFv3 routes into BGP using a route map with the redistribution statement

Scenario

In this scenario, BGP_SW1 is to only advertise External OSPFv3 route 2001:db8:11::/64 into BGP.



Objective

- Redistribute selected OSPFv3 routes into BGP only.
- OSPFv3 External Routes 2001:db8:12::/64 and 2001:db8:13::/64 are not to be redistributed into BGP.

Prerequisite

- OSPFv3 adjacency must be formed between OSPF_SW1 and BGP_SW1.
- BGP_SW1 knows of external networks advertised by OSPFv3 neighbor 2.2.2.2 (OSPF_SW1).

```
BGP_SW1#show running-config interface vlan10
!
interface vlan10
 ipv6 address 2001:db8:10::1/64
 ipv6 enable
 ipv6 router ospf area 0
!

BGP_SW1#show ipv6 ospf neighbor
OSPFv3 Process (*null*)
Neighbor ID     Pri   State           Dead Time   Interface  Instance ID
2.2.2.2           1   Full/DR         00:00:33    vlan10     0
```

```
BGP_SW1#show running-config router ipv6 ospf
!
router ipv6 ospf
 router-id 1.1.1.1
 no bgp default ipv4-unicast
!


BGP_SW1#show ipv6 ospf route
OSPFv3 Process (*null*)
Codes: C - connected, D - Discard, O - OSPF, IA - OSPF inter area
       E1 - OSPF external type 1, E2 - OSPF external type 2

   Destination       Metric Next-hop
C  2001:db8:10::/64  1      directly connected, vlan10, Area 0.0.0.0
E2 2001:db8:11::/64  1/20   via fe80::200:cdff:fe28:7e6, vlan10
E2 2001:db8:12::/64  1/20   via fe80::200:cdff:fe28:7e6, vlan10
E2 2001:db8:13::/64  1/20   via fe80::200:cdff:fe28:7e6, vlan10
```

1. Create a Prefix List or ACL on BGP_SW1 to permit the 2001:db8:11::/64 prefix.

Using an ACL:

```
BGP_SW1(config)#ipv6 access-list standard OSPFv3_FILTER_LIST permit
   2001:db8:11::/64
```

OR using a Prefix List:

```
BGP_SW1(config)#ipv6 prefix-list OSPFv3_FILTER_LIST seq 10 permit
   2001:db8:11::/64
```

2. Create a route map to match the ACL or Prefix List.

Using an ACL:

```
BGP_SW1(config)#route-map OSPFv3_REDISTRIBUTE permit 10
BGP_SW1(config-route-map)#match ipv6 address OSPFv3_FILTER_LIST
```

OR using a Prefix List:

```
BGP_SW1(config)#route-map OSPFv3_REDISTRIBUTE permit 10
BGP_SW1(config-route-map)#match ipv6 address prefix-list
   OSPFv3_FILTER_LIST
```

3. Redistribute OSPFv3 networks permitted by the router into BGP.

```
BGP_SW1(config)#router bgp 100
BGP_SW1(config-router)#address-family ipv6
BGP_SW1(config-router-af)#redistribute ospf route-map
   OSPFv3_REDISTRIBUTE
```

Verify by viewing the:

- BGP table on BGP_SW2 to check for the advertised prefix.
- IPv6 routing table on BGP_SW2.
- Route map on BGP_SW1 to verify only the selected prefix has been redistributed.
- Prefix list counter for bgpd on BGP_SW1.

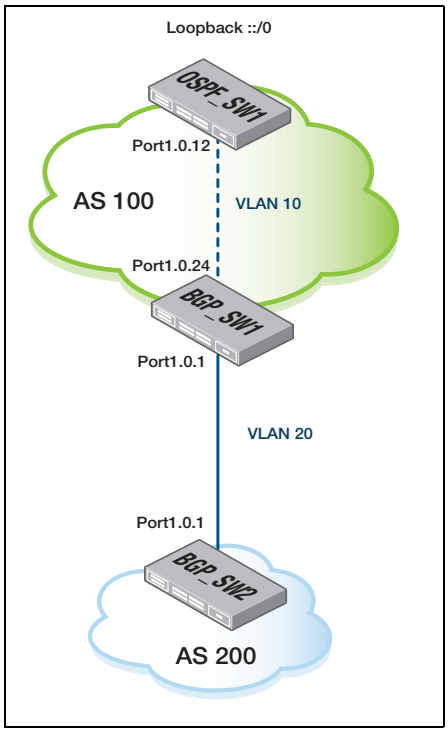
```
BGP_SW2#show bgp ipv6
BGP table version is 26, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:11::/64 2001:db8:1::1           20             0 100 ?

Number of prefixes 1


BGP_SW2#show ipv6 route
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
       IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
Timers: Uptime

S     2001:db8:1::1/128 [1/0] via fe80::200:cdff:fe2a:eaf1, vlan20, 01:00:10
C     2001:db8:2::/64 via ::, lo, 01:00:10
B     2001:db8:11::/64 [20/20] via 2001:db8:1::1 (recursive via
fe80::200:cdff:fe2a:eaf1), vlan20, 00:01:07
C     2001:db8:20::/64 via ::, vlan20, 01:00:02
C     fe80::/64 via ::, vlan20, 01:00:02
C     fe80::/64 via ::, lo, 01:00:10
```

```
BGP_SW1#show bgp ipv6 route-map OSPFv3_REDISTRIBUTE
BGP table version is 56, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network               Next Hop            Metric LocPrf Weight Path
*> 2001:db8:11::/64 fe80::200:cdff:fe28:7e6       20        32768  ?

Number of prefixes 1
```

```
BGP_SW1#show ipv6 prefix-list detail
Prefix-list with the last deletion/insertion: OSPFv3_FILTER_LIST
ipv6 prefix-list OSPFv3_FILTER_LIST:
   count: 1, range entries: 0, sequences: 10 - 10
    ripd:
     seq 10 permit 2001:db8:11::/64 (hit count: 0, refcount: 0)
    ripngd:
     seq 10 permit 2001:db8:11::/64 (hit count: 0, refcount: 0)
    ospfd:
     seq 10 permit 2001:db8:11::/64 (hit count: 0, refcount: 0)
    ospf6d:
     seq 10 permit 2001:db8:11::/64 (hit count: 0, refcount: 0)
    bgpd:
     seq 10 permit 2001:db8:11::/64 (hit count: 1, refcount: 19)
```

Using a route map with BGP default-originate to advertise a default route conditionally

Scenario



Objective

- Advertise a default route into BGP only if a default route ::/0 is advertised by OSPF_SW1 to BGP_SW1.
- Conditionally, IPv6 default route must be present in BGP_SW1 routing table for the route to be advertised.

Prerequisite

- OSPFv3 adjacency must be formed between OSPF_SW1 and BGP_SW1.
- OSPF_SW1 advertises IPv6 default route via OSPFv3 to BGP_SW1.

```
BGP_SW1#show running-config router ipv6 ospf
!
router ipv6 ospf
 router-id 1.1.1.1
!


BGP_SW1#show running-config interface vlan10
!
interface vlan10
 ipv6 address 2001:db8:10::1/64
 ipv6 router ospf area 0
!
```

```
BGP_SW1#show ipv6 ospf neighbor
OSPFv3 Process (*null*)
Neighbor ID     Pri   State           Dead Time   Interface  Instance ID
2.2.2.2           1   Full/DR         00:00:33    vlan10     0


BGP_SW1#show ipv6 ospf route
OSPFv3 Process (*null*)
Codes: C - connected, D - Discard, O - OSPF, IA - OSPF inter area
       E1 - OSPF external type 1, E2 - OSPF external type 2

   Destination       Metric Next-hop
E2 ::/0              1/20   via fe80::200:cdff:fe28:7e6, vlan10
C  2001:db8:10::/64  1      directly connected, vlan10, Area 0.0.0.0
```

1. Create an ACL to match the IPv6 default route being advertised by OSPFv3 neighbor OSPF_SW1.

- The exact-match parameter is required as we want specifically the all zeros IPv6 default route address.
- If this parameter is not set, the switch considers the route matching any prefix.

```
BGP_SW1(config)#ipv6 access-list standard OSPFv3_LIST permit ::/0
   exact-match
```

2. Create a route map to match on the OSPFv3_LIST.

- In this case we want to match the exact IPv6 default route using the route map.

```
BGP_SW1(config)#route-map DEFAULT_ORIGINATE permit 10
BGP_SW1(config-route-map)#match ipv6 address OSPFv3_LIST
```

3. Redistribute OSPFv3 networks permitted by the router into BGP.

- A default route will be advertised to BGP neighbor 2001:db8:2::1 (BGP_SW2) if the route map matches on an existing IPv6 default route.
- If no IPv6 default route exists in BGP_SW1's routing table, no default route will be advertised to BGP neighbor BGP_SW2.

```
BGP_SW1(config)#router bgp 100
BGP_SW1(config-router)#address-family ipv6
BGP_SW1(config-router-af)#neighbor 2001:db8:2::1 default-originate
   route-map DEFAULT_ORIGINATE
```

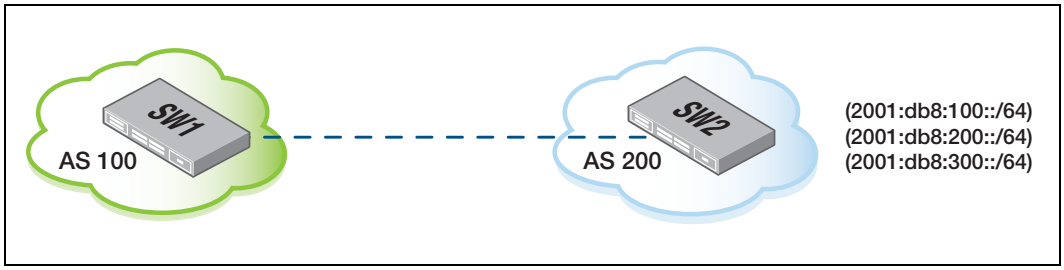
Verify by viewing the:

- BGP table on BGP_SW2.
- IPv6 routing table on BGP_SW2.

```
BGP_SW2#show bgp ipv6
BGP table version is 2, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> ::/0             2001:db8:1::1            0             0 100 i

Number of prefixes 1
```

```
BGP_SW2#show ipv6 route database
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
       IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
       > - selected route, * - FIB route, p - stale info
Timers: Uptime

B  *> ::/0 [20/0] via 2001:db8:1::1 (recursive via fe80::200:cdff:fe2a:eaf1),
vlan20, 00:02:36
S  *> 2001:db8:1::1/128 [1/0] via fe80::200:cdff:fe2a:eaf1, vlan20, 02:37:28
C  *> 2001:db8:2::/64 via ::, lo, 02:37:28
S     2001:db8:3::1/128 [1/0] via fe80::200:cdff:fe2a:e635, vlan40 inactive,
02:37:28
C  *> 2001:db8:20::/64 via ::, vlan20, 02:37:20
C  *  fe80::/64 via ::, vlan20, 02:37:20
C  *> fe80::/64 via ::, lo, 02:37:28
```

Using redistribution to dynamically advertise RIPng routes into BGP



Use

Dynamically advertise RIPng routes into BGP.

Objective

Redistribute the RIPng routes in AS 200 into BGP and advertise the prefixes from SW2 in AS 200 to SW1 in AS 100.

Check SW2 has learned the RIPng routes from SW3 in AS 200:

```
SW2#show ipv6 route database
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
> - selected route, * - FIB route, p - stale info
Timers: Uptime
R 2001:db8:10::/64 [0/1] via ::, vlan10, 00:25:05
C *> 2001:db8:10::/64 via ::, vlan10, 00:25:05
R 2001:db8:20::/64 [0/1] via ::, vlan20, 00:25:05
C *> 2001:db8:20::/64 via ::, vlan20, 00:25:05
R *> 2001:db8:100::/64 [120/2] via fe80::eecd:6dff:fe48:e54e, vlan20, 00:25:00
R *> 2001:db8:200::/64 [120/2] via fe80::eecd:6dff:fe48:e54e, vlan20, 00:25:00
R *> 2001:db8:300::/64 [120/2] via fe80::eecd:6dff:fe48:e54e, vlan20, 00:25:00
C *> fe80::/64 via ::, vlan10, 00:25:05
C * fe80::/64 via ::, vlan20, 00:25:12
C * fe80::/64 via ::, vlan10, 00:25:12
```

1. Setup the redistribution of RIPng routes into BGP.

```
SW2(config)#router bgp 200
SW2(config-router)#address-family ipv6
SW2(config-router-af)#redistribute rip
```

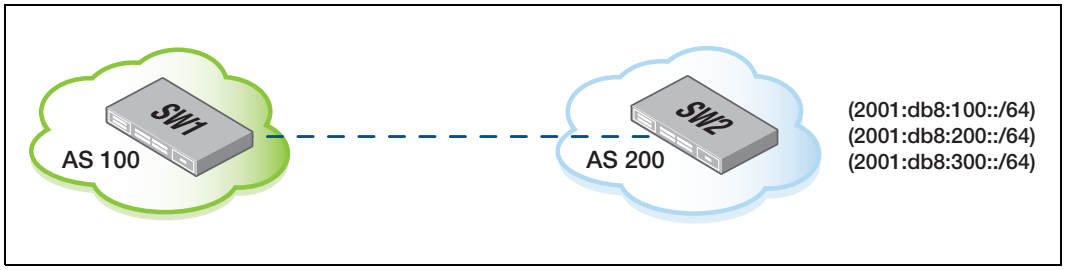
Verify

- SW2 is advertising the RIPng routes into BGP. (Notice the "?" incomplete Origin)
- By viewing the BGP table on SW1.

```
SW2#show bgp ipv6
BGP table version is 3, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network            Next Hop                Metric LocPrf Weight Path
*> 2001:db8:100::/64
                   fe80::eecd:6dff:fe48:e54e
                                                2             32768 ?
*> 2001:db8:200::/64
                   fe80::eecd:6dff:fe48:e54e
                                                2             32768 ?
*> 2001:db8:300::/64
                   fe80::eecd:6dff:fe48:e54e
                                                2             32768 ?
Number of prefixes 3
```

```
SW1#show bgp ipv6
BGP table version is 4, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network              Next Hop              Metric LocPrf Weight Path
*> 2001:db8:100::/64
                    2001:db8:10::2(fe80::eecd:6dff:fe20:c073)
                                                2             0 200 ?
*> 2001:db8:200::/64
                    2001:db8:10::2(fe80::eecd:6dff:fe20:c073)
                                                2             0 200 ?
*> 2001:db8:300::/64
                    2001:db8:10::2(fe80::eecd:6dff:fe20:c073)
                                                2             0 200 ?
Number of prefixes 3
```

Using a route map to redistribute specific RIPng routes into BGP



Use

Dynamically advertise specific RIPng routes into BGP.

Objective

- Redistribute the 2001:db8:200::/64 prefix learned via RIPng into BGP.
- Other prefixes should not be redistributed into BGP.

Check SW2 has learned the RIPng routes from SW3 in AS 200.

```
SW2#show ipv6 route database
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
> - selected route, * - FIB route, p - stale info
Timers: Uptime
R 2001:db8:10::/64 [0/1] via ::, vlan10, 15:43:21
C *> 2001:db8:10::/64 via ::, vlan10, 15:43:21
R 2001:db8:20::/64 [0/1] via ::, vlan20, 15:43:21
C *> 2001:db8:20::/64 via ::, vlan20, 15:43:21
R *> 2001:db8:100::/64 [120/2] via fe80::eecd:6dff:fe48:e54e, vlan20, 15:43:16
R *> 2001:db8:200::/64 [120/2] via fe80::eecd:6dff:fe48:e54e, vlan20, 15:43:16
R *> 2001:db8:300::/64 [120/2] via fe80::eecd:6dff:fe48:e54e, vlan20, 15:43:16
C *> fe80::/64 via ::, vlan10, 15:43:21
C * fe80::/64 via ::, vlan20, 15:43:28
C * fe80::/64 via ::, vlan10, 15:43:28
```

1. Setup an IPv6 ACL to permit only the 2001:db8:200::/64 prefix.

```
SW2(config)#ipv6 access-list standard REDISITRIBUTION_LIST permit
   2001:db8:200::/64
SW2(config)#ipv6 access-list standard REDISITRIBUTION_LIST deny any
```

2. Setup the route map to match on the IPv6 ACL.

```
SW2(config)#route-map REDISTRIBUTION_MAP permit 10
SW2(config-route-map)#match ipv6 address REDISTRIBUTION_LIST
```

3. Setup the redistribution of RIPng routes.

```
SW2(config)#router bgp 200
SW2(config-router)#address-family ipv6
SW2(config-router-af)#redistribute rip route-map REDISTRIBUTION_MAP
```

Verify by viewing:

- SW2 is advertising the RIPng route 2001:db8:200::/64 into BGP. (Notice the "?" incomplete Origin).
- The BGP routes matching the route map on SW2.
- The BGP table on SW1.

```
SW2#show bgp ipv6
BGP table version is 930, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network            Next Hop            Metric LocPrf Weight Path
*> 2001:db8:200::/64
                  fe80::eecd:6dff:fe48:e54e
                                                2             32768  ?
Number of prefixes 1
```

```
SW2#show bgp ipv6 route-map REDISTRIBUTION_MAP
BGP table version is 931, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network            Next Hop            Metric LocPrf Weight Path
*> 2001:db8:200::/64
                   fe80::eecd:6dff:fe48:e54e
                                                2              32768  ?
Number of prefixes 1

SW1#show bgp ipv6
BGP table version is 2, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network            Next Hop            Metric LocPrf Weight Path
*> 2001:db8:200::/64
                   2001:db8:10::2(fe80::eecd:6dff:fe20:c073)
                                                2              0  200 ?
Number of prefixes 1
```

This chapter covers the following topics:

- Using regular expressions to filter prefixes based on AS path
- Using regular expression to deny prefixes originating from a specific AS in outgoing updates
- Using regular expression to only deny networks originating from a specific AS
- Using regular expression to only permit networks originating from a neighbor AS
- Using regular expression to only permit networks originating from a specific AS
- Using regular expression to only permit networks passing through a specific AS
- Using distribute lists to filter routing updates - Inbound method
- Using distribute lists to filter routing updates - Outbound method
- Using BGP Outbound Route Filtering to dynamically filter outbound updates
- Using Prefix lists to filter routing updates - Inbound method
- Using Prefix lists to filter routing updates - Outbound method
- Using the BGP well-known community attribute (no-advertise) to perform route filtering

CHAPTER 4

Route Filtering

You can filter BGP and BGP4+ advertisements in various ways with neighbors, including:

1. Autonomous System path filters applied using the **neighbor filter-list** command matching on an **ip as-path access-list** (BGP and BGP4+).
2. Distribute lists applied using the **neighbor distribute-list** command, matching on an access-list (BGP and BGP4+).
3. Prefix list applied using the **neighbor prefix-list** command, matching on a prefix list (BGP and BGP4+).
4. Route maps applied using the **neighbor route-map** command, matching either an access-list, prefix-list or an as-path access-list (BGP and BGP4+).

If you want to restrict routing information that is learned or advertised, you can selectively filter BGP routing updates to and from neighbors. You can define an access list or a prefix list and apply it to the updates.

Using regular expressions to filter prefixes based on AS path

EXPRESSION	DEFINITION	EXAMPLE OF USE
^100$	Indicates origination from neighbor AS 100.	Prefixes which have AS 100 at the beginning and the end of the AS_PATH are matched.
_100_	Passes through AS 100.	Prefixes which have AS 100 in the AS_PATH are matched.
_100$	Indicates an origin of AS 100.	Prefixes which have AS 100 at the end of the AS_PATH are matched.
^100 .*	Indicates transmission from AS 100.	Prefixes which have AS 100 at the beginning of the AS_PATH are matched.
.*	Indicates from any AS.	Prefixes with any AS in the AS_PATH are matched. (Equivalent of "any" statement in ACLs).

Note for regular expression matching

- Beginning of AS_PATH = newest AS added to the AS path.
- End of AS_PATH = first (originating) AS in the AS path.

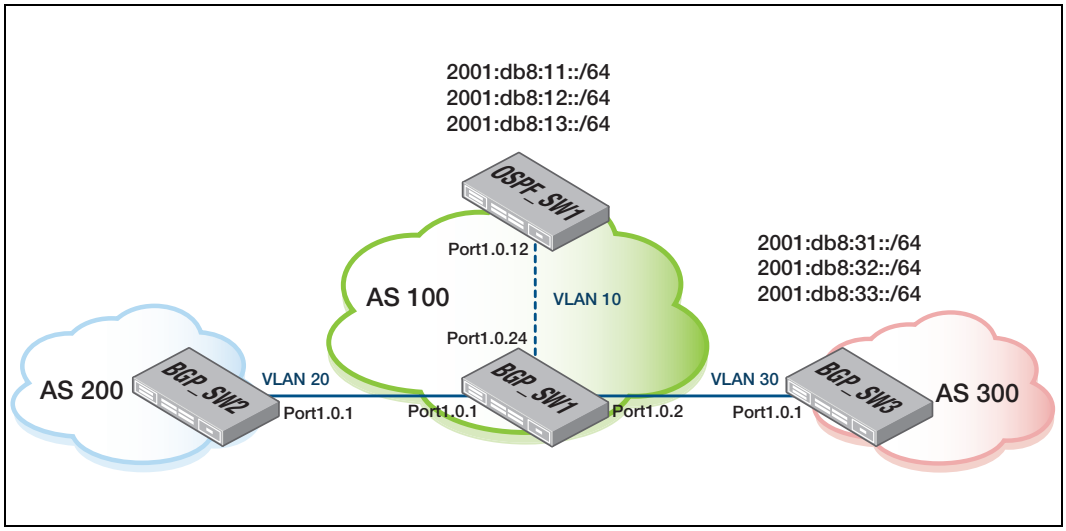
AS Path example

- AS Path 200 300 400:
- 200 is the beginning of the AS_PATH, 400 is the end and originating AS in the AS_PATH.

Using regular expression to deny prefixes originating from a specific AS in outgoing updates

Scenario

- BGP_SW1 resides in AS 100.
- BGP_SW2 resides in AS 200.
- BGP_SW3 resides in AS 300.



Objective

- Prefixes transiting from AS 300 are to be filtered (removed) from outgoing updates when advertised from AS 100 to AS 200.

To achieve this objective, an AS path ACL is to be applied to BGP_SW1 to filter prefixes originating from AS 300 in the outgoing updates to AS 200.

View of BGP_SW2's BGP table prior to route filtering

```
BGP_SW2#show bgp ipv6
BGP table version is 7, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:11::/64 2001:db8:1::1            0             0 100 i
*> 2001:db8:12::/64 2001:db8:1::1            0             0 100 i
*> 2001:db8:13::/64 2001:db8:1::1            0             0 100 i
*> 2001:db8:31::/64 2001:db8:1::1            0             0 100 300 i
*> 2001:db8:32::/64 2001:db8:1::1            0             0 100 300 i
*> 2001:db8:33::/64 2001:db8:1::1            0             0 100 300 i
Number of prefixes 6
```

1. Create an AS path containing a regular expression ACL, to deny prefixes being advertised to BGP_SW2 in AS 200 that originate from AS 300.

```
BGP_SW1(config)#ip as-path access-list AS_LIST deny ^300$
BGP_SW1(config)#ip as-path access-list AS_LIST permit .*
```

2. Apply the filter list to the 2001:db8:2::1 neighbor to filter prefixes from AS 300 being advertised.

```
BGP_SW1(config)#router bgp 100
BGP_SW1(config-router)#address-family ipv6
BGP_SW1(config-router-af)#neighbor 2001:db8:2::1 filter-list AS_LIST out
```

3. Clear the BGP session to neighbor 2001:db8:2::1 (BGP_SW2 in AS 200) to apply the change.

```
BGP_SW1#clear bgp ipv6 2001:db8:2::1
```

Verify by viewing the:

- BGP table on BGP_SW2 in AS 200.
- BGP neighbor session on BGP_SW1 in AS 100 (outgoing update AS path filter list).
- AS path ACL on BGP_SW1.
- Filter-list on BGP_SW1.

```
BGP_SW2#show bgp ipv6
BGP table version is 18, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:11::/64 2001:db8:1::1            0             0 100 i
*> 2001:db8:12::/64 2001:db8:1::1            0             0 100 i
*> 2001:db8:13::/64 2001:db8:1::1            0             0 100 i
 Number of prefixes 3
```

```
BGP_SW1#show bgp ipv6 neighbors 2001:db8:2::1
BGP neighbor is 2001:db8:2::1, remote AS 200, local AS 100, external link
  BGP version 4, remote router ID 2.2.2.2
  BGP state = Established, up for 00:13:26
  Last read 00:13:26, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    4-Octet ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 158 messages, 0 notifications, 0 in queue
  Sent 162 messages, 2 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
  Update source is lo
For address family: IPv6 Unicast
  BGP table version 25, neighbor version 25
  Index 1, Offset 0, Mask 0x2
  Community attribute sent to this neighbor (both)
  Outbound path policy configured
  Outgoing update AS path filter list is *AS_LIST
  0 accepted prefixes
  3 announced prefixes

 Connections established 3; dropped 2
  External BGP neighbor may be up to 2 hops away.
Local host: 2001:db8:1::1, Local port: 59208
Foreign host: 2001:db8:2::1, Foreign port: 179
Nexthop: 1.1.1.1
Nexthop global: 2001:db8:1::1
Nexthop local: ::
BGP connection: non shared network
Last Reset: 00:13:27, due to BGP Notification sent
Notification Error Message: (Cease/Other Configuration Change.)
Number of prefixes 3
```

```
BGP_SW1#show bgp ipv6 neighbors 2001:db8:2::1
BGP neighbor is 2001:db8:2::1, remote AS 200, local AS 100, external link
  BGP version 4, remote router ID 2.2.2.2
  BGP state = Established, up for 00:13:26
  Last read 00:13:26, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    4-Octet ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 158 messages, 0 notifications, 0 in queue
  Sent 162 messages, 2 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
  Update source is lo
For address family: IPv6 Unicast
  BGP table version 25, neighbor version 25
  Index 1, Offset 0, Mask 0x2
  Community attribute sent to this neighbor (both)
  Outbound path policy configured
  Outgoing update AS path filter list is *AS_LIST
  0 accepted prefixes
  3 announced prefixes

 Connections established 3; dropped 2
  External BGP neighbor may be up to 2 hops away.
Local host: 2001:db8:1::1, Local port: 59208
Foreign host: 2001:db8:2::1, Foreign port: 179
Nexthop: 1.1.1.1
Nexthop global: 2001:db8:1::1
Nexthop local: ::
BGP connection: non shared network
Last Reset: 00:13:27, due to BGP Notification sent
Notification Error Message: (Cease/Other Configuration Change.)
```

```
BGP_SW1#show ip as-path-access-list
AS path access list AS_LIST
    deny ^300$
    permit .*

BGP_SW1#show bgp ipv6 filter-list AS_LIST
BGP table version is 32, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:11::/64 ::                       100          32768 i
*> 2001:db8:12::/64 ::                       100          32768 i
*> 2001:db8:13::/64 ::                       100          32768 i

Number of prefixes 3
```

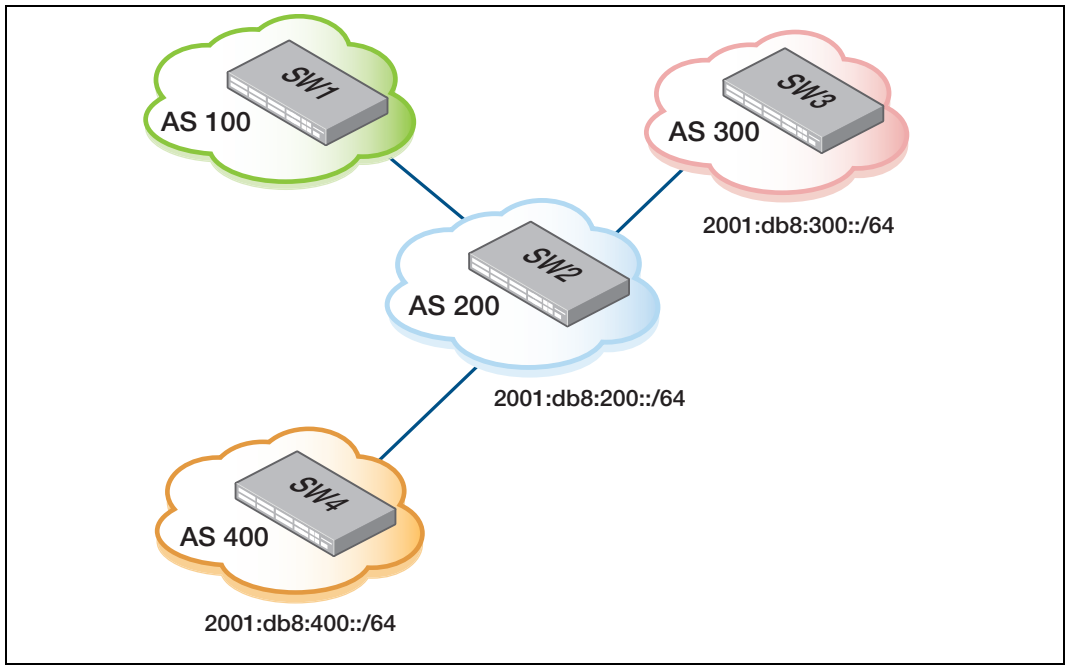
Configuration on BGP_SW1

```
BGP_SW1#
!
hostname BGP_SW1
!
vlan database
 vlan 10,20,30 state enable
!
interface port1.0.1
 switchport access vlan 20
!
interface port1.0.2
 switchport access vlan 30
!
interface port1.0.24
 switchport access vlan 10
!
interface lo
 ipv6 address 2001:db8:1::1/64
!
interface vlan10
 ipv6 address 2001:db8:10::1/64
 ipv6 enable
 ipv6 router ospf area 0
!
interface vlan20
 ipv6 address 2001:db8:20::1/64
!
interface vlan30
 ipv6 address 2001:db8:30::1/64
!
ipv6 ospf display route single-line
!
router ipv6 ospf
 router-id 1.1.1.1
!
router bgp 100
 bgp router-id 1.1.1.1
 no bgp default ipv4-unicast
 neighbor 2001:db8:2::1 remote-as 200
 neighbor 2001:db8:2::1 ebgp-multihop 2
 neighbor 2001:db8:2::1 update-source lo
 neighbor 2001:db8:2::1 activate
 neighbor 2001:db8:3::1 remote-as 300
 neighbor 2001:db8:3::1 ebgp-multihop 2
 neighbor 2001:db8:3::1 update-source lo
 neighbor 2001:db8:3::1 activate
 !
 address-family ipv6
 network 2001:db8:11::/64
 network 2001:db8:12::/64
 network 2001:db8:13::/64
 neighbor 2001:db8:2::1 activate
 neighbor 2001:db8:2::1 filter-list AS_LIST out
 neighbor 2001:db8:3::1 activate
 exit-address-family
!
ip as-path access-list AS_LIST deny ^300$
ip as-path access-list AS_LIST permit .*
!
ipv6 route 2001:db8:2::1/128 fe80::209:41ff:fefb:c323 vlan20
ipv6 route 2001:db8:3::1/128 fe80::200:cdff:fe2a:e635 vlan30
!
end
```

Using regular expression to only deny networks originating from a specific AS

Scenario

Only deny networks that originate from remote AS 300.



Use

Perform route filtering based on the AS_PATH attribute.

Objective

On SW1, apply route filtering using the AS_PATH attribute to deny prefixes originating from AS 300 only.

Example

SW1's BGP table prior to route filtering:

```
SW1#show bgp ipv6
BGP table version is 3, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network          Next Hop                    Metric LocPrf Weight Path
*> 2001:db8:200::/64
                 2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                                  0             0    200 i
*> 2001:db8:300::/64
                 2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                                  0             0    200 300 i
*> 2001:db8:400::/64
                 2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                                  0             0    200 400 i
Number of prefixes 3
```

1. Create an AS path ACL on SW1 to deny prefixes that originate from remote AS 300.

- Prefixes originating from AS 300 are denied.
- Prefixes from any other AS are permitted.

```
SW1(config)#ip as-path access-list PERMIT_AS_LIST deny _300$
SW1(config)#ip as-path access-list PERMIT_AS_LIST permit .*
```

2. Create a route map to match on the AS path ACL.

```
SW1(config)#route-map PERMIT_AS_MAP permit 10
SW1(config-route-map)#match as-path PERMIT_AS_LIST
SW1(config-route-map)#exit
```

3. Assign the route map to the peering to apply the route filtering.

```
SW1(config)#router bgp 100
SW1(config-router)#address-family ipv6
SW1(config-router-af)#neighbor 2001:db8:10::2 route-map PERMIT_AS_MAP in
```

4. Clear the BGP session to the eBGP neighbor in AS 200 to apply the change.

```
SW1#clear bgp ipv6 2001:db8:10::2
```

Verify by:

- Viewing the BGP table.
- Viewing the successful matches to the route map.
- Using the show bgp ipv6 neighbors command to see the accepted prefix counter and the route map applied to the incoming advertisements.

```
SW1#show bgp ipv6
BGP table version is 2, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network             Next Hop                  Metric LocPrf Weight Path
*> 2001:db8:200::/64
                    2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                                   0             0   200 i
*> 2001:db8:400::/64
                    2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                                   0             0   200 400 i
Number of prefixes 2
```

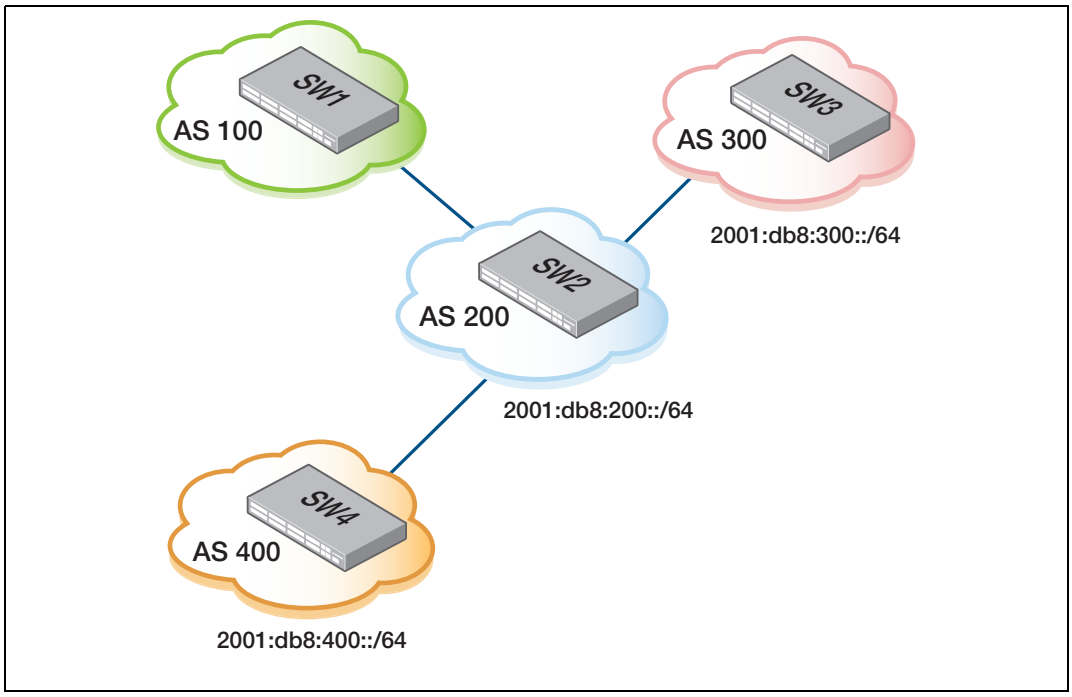
```
SW1#show bgp ipv6 route-map PERMIT_AS_MAP
BGP table version is 2, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network             Next Hop                          Metric LocPrf Weight Path
*> 2001:db8:200::/64
                        2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                                            0   0        200 i
*> 2001:db8:400::/64
                        2001:db8:10::2(fe80::eecd:6dff:fe20:c26b
)                                                           0   0     200 400 i
Number of prefixes 2
```

```
SW1#show bgp ipv6 neighbors
BGP neighbor is 2001:db8:10::2, remote AS 200, local AS 100, external link
  BGP version 4, remote router ID 2.2.2.2
  BGP state = Established, up for 00:00:36
  Last read 00:00:36, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    4-Octet ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 392 messages, 0 notifications, 0 in queue
  Sent 352 messages, 11 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
For address family: IPv6 Unicast
 BGP table version 2, neighbor version 2
 Index 1, Offset 0, Mask 0x2
 Community attribute sent to this neighbor (both)
 Inbound path policy configured
  Route map for incoming advertisements is *PERMIT_AS_MAP
 2 accepted prefixes
 0 announced prefixes

Connections established 10; dropped 9
Local host: 2001:db8:10::1, Local port: 60767
Foreign host: 2001:db8:10::2, Foreign port: 179
Nexthop: 1.1.1.1
Nexthop global: 2001:db8:10::1
Nexthop local: fe80::eecd:6dff:fe52:b789
BGP connection: shared network
Last Reset: 00:00:36, due to BGP Notification sent
Notification Error Message: (Cease/Other Configuration Change.)
```

Using regular expression to only permit networks originating from a neighbor AS

Scenario

Only permit networks that originated in AS 200.



Use

Perform route filtering based on the AS_PATH attribute.

Objective

On SW1 apply route filtering using the AS_PATH attribute to allow only prefixes originating from AS 200.

Example: SW1's BGP table prior to route filtering

```
SW1#show bgp ipv6
BGP table version is 22, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network          Next Hop                    Metric LocPrf Weight Path
*> 2001:db8:200::/64
                 2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                                  0             0     200 i
*> 2001:db8:300::/64
                 2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                                  0             0  200 300 i
*> 2001:db8:400::/64
                 2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                                  0             0  200 400 i

Number of prefixes 3
```

1. Create an AS path ACL on SW1 to permit prefixes with only AS_200 in the AS_PATH.

- The AS path must start and end with 200.

```
SW1(config)#ip as-path access-list PERMIT_AS_LIST permit ^200$
```

2. Create a route map to match on the AS path ACL.

```
SW1(config)#route-map PERMIT_AS_MAP permit 10
SW1(config-route-map)#match as-path PERMIT_AS_LIST
SW1(config-route-map)#exit
```

3. Assign the route map to the peering to apply the route filtering.

```
SW1(config)#router bgp 100
SW1(config-router)#address-family ipv6
SW1(config-router-af)#neighbor 2001:db8:10::2 route-map PERMIT_AS_MAP in
```

4. Clear the BGP session on the eBGP neighbor in AS 200 to apply the change.

```
SW1#clear bgp ipv6 2001:db8:10::2
```

Verify by:

- Viewing the BGP table. (Only the prefix originating from AS 200 is now present)
- Viewing the successful matches to the route map.
- Using the show bgp ipv6 neighbors command to see the accepted prefix counter and the route map applied to the incoming advertisements.

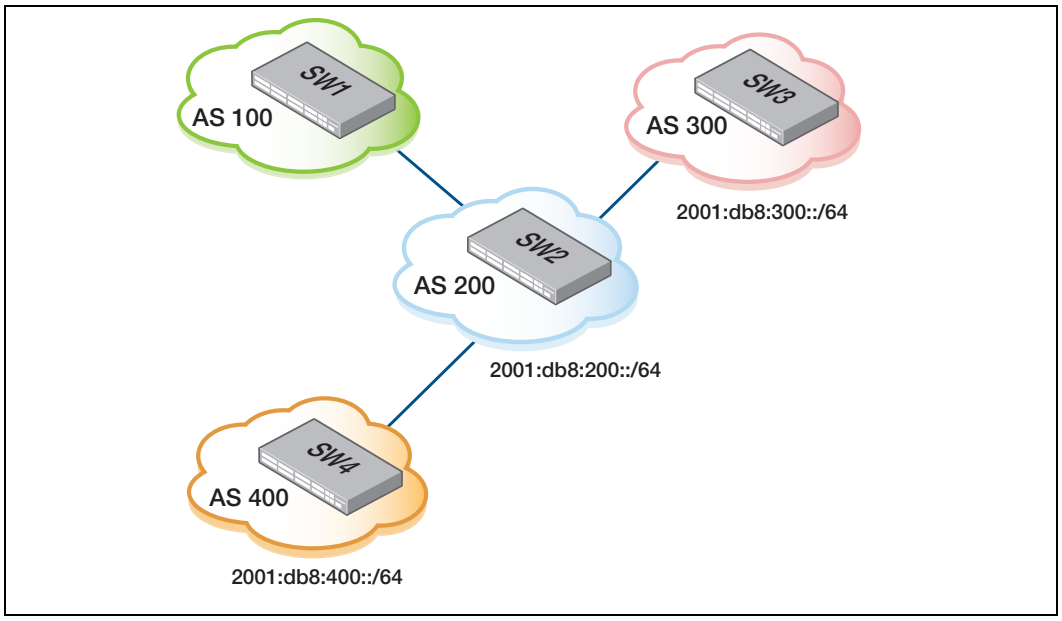
```
SW1#show bgp ipv6
BGP table version is 28, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:200::/64
                 2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                          0             0 200 i
Number of prefixes 1

SW1#show bgp ipv6 route-map PERMIT_AS_MAP
BGP table version is 28, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network           Next Hop            Metric LocPrf Weight Path
*> 2001:db8:200::/64
                  2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                          0             0        200 i
Number of prefixes 1
```

```
SW1#show bgp ipv6 neighbors
BGP neighbor is 2001:db8:10::2, remote AS 200, local AS 100, external link
  BGP version 4, remote router ID 2.2.2.2
  BGP state = Established, up for 00:03:39
  Last read 00:03:39, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
   Route refresh: advertised and received (old and new)
   4-Octet ASN Capability: advertised and received
   Address family IPv6 Unicast: advertised and received
 Received 76 messages, 1 notifications, 0 in queue
 Sent 73 messages, 2 notifications, 0 in queue
 Route refresh request: received 0, sent 0
 Minimum time between advertisement runs is 30 seconds
For address family: IPv6 Unicast
 BGP table version 28, neighbor version 28
 Index 1, Offset 0, Mask 0x2
 Community attribute sent to this neighbor (both)
 Inbound path policy configured
 Route map for incoming advertisements is *PERMIT_AS_MAP
 1 accepted prefixes
 0 announced prefixes

 Connections established 3; dropped 2
Local host: 2001:db8:10::1, Local port: 179
Foreign host: 2001:db8:10::2, Foreign port: 41270
Nexthop: 1.1.1.1
Nexthop global: 2001:db8:10::1
Nexthop local: fe80::eecd:6dff:fe52:b789
BGP connection: shared network
Last Reset: 00:03:39, due to BGP Notification sent
Notification Error Message: (Cease/Administratively Reset.)
```

Using regular expression to only permit networks originating from a specific AS

Scenario

Only permit networks that originate from remote AS 300.



Use

Perform route filtering based on the AS_Path attribute.

Objective

On SW1 apply route filtering using the AS-PATH attribute to permit prefixes originating from AS 300 only.

Example

SW1's BGP table prior to route filtering.

```
SW1#show bgp ipv6
BGP table version is 6, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network               Next Hop                 Metric LocPrf Weight Path
*> 2001:db8:200::/64
                      2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                                    0             0  200 i
*> 2001:db8:300::/64
                      2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                                    0             0  200 300 i
*> 2001:db8:400::/64
                      2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                                    0             0  200 400 i
Number of prefixes 3
```

1. Create AS path ACL on SW1 to permit prefixes that originate from remote AS 300.

The prefix must originate from AS 300 to be permitted.

```
SW1(config)#ip as-path access-list PERMIT_AS_LIST permit _300$
```

2. Create a route map to match on the AS path ACL.

```
SW1(config)#route-map PERMIT_AS_MAP permit 10
SW1(config-route-map)#match as-path PERMIT_AS_LIST
SW1(config-route-map)#exit
```

3. Assign the route map to the peering to apply the route filtering.

```
SW1(config)#router bgp 100
SW1(config-router)#address-family ipv6
SW1(config-router-af)#neighbor 2001:db8:10::2 route-map PERMIT_AS_MAP
   in
```

4. Clear the BGP session to the eBGP neighbor in AS 200 to apply the change.

```
SW1#clear bgp ipv6 2001:db8:10::2
```

Verify by:

- Viewing the BGP table. (Only the prefix originating from AS 300 is now present.)
- Viewing the successful matches to the route map.
- Using the show bgp ipv6 neighbors command to view the accepted prefix counter and the route map applied to the incoming advertisements.

```
SW1#show bgp ipv6
BGP table version is 2, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale

Origin codes: i - IGP, e - EGP, ? - incomplete
Network             Next Hop             Metric LocPrf Weight Path
*> 2001:db8:300::/64
                    2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                              0             0    200 300 i
Number of prefixes 1
```

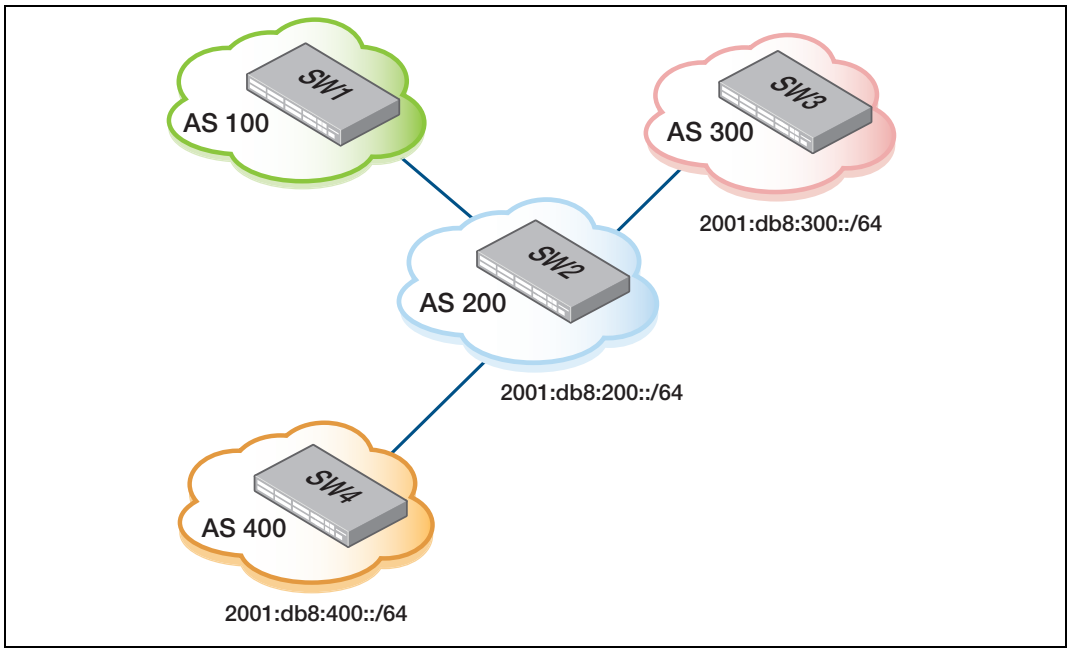
```
SW1#show bgp ipv6 route-map PERMIT_AS_MAP
BGP table version is 2, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale

Origin codes: i - IGP, e - EGP, ? - incomplete
Network              Next Hop             Metric LocPrf Weight Path
*> 2001:db8:300::/64
                     2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                               0            0   200 300 i
Number of prefixes 1
```

```
SW1#show bgp ipv6 neighbors
BGP neighbor is 2001:db8:10::2, remote AS 200, local AS 100, external link
  BGP version 4, remote router ID 2.2.2.2
  BGP state = Established, up for 00:00:40
  Last read 00:00:40, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
   Route refresh: advertised and received (old and new)
   4-Octet ASN Capability: advertised and received
   Address family IPv6 Unicast: advertised and received
 Received 318 messages, 0 notifications, 0 in queue
 Sent 293 messages, 7 notifications, 0 in queue
 Route refresh request: received 0, sent 0
 Minimum time between advertisement runs is 30 seconds
For address family: IPv6 Unicast
 BGP table version 2, neighbor version 2
 Index 1, Offset 0, Mask 0x2
 Community attribute sent to this neighbor (both)
 Inbound path policy configured
 Route map for incoming advertisements is *PERMIT_AS_MAP
 1 accepted prefixes
 0 announced prefixes

Connections established 6; dropped 5
Local host: 2001:db8:10::1, Local port: 179
Foreign host: 2001:db8:10::2, Foreign port: 46459
Nexthop: 1.1.1.1
Nexthop global: 2001:db8:10::1
Nexthop local: fe80::eecd:6dff:fe52:b789
BGP connection: shared network
Last Reset: 00:00:40, due to BGP Notification sent
Notification Error Message: (Cease/Other Configuration Change.)
```

Using regular expression to only permit networks passing through a specific AS

Scenario

Only permit networks that originated or have passed through AS 300.



Use

Perform route filtering based on the AS_PATH attribute.

Objective

On SW1 apply route filtering using the AS_PATH attribute to allow only prefixes that originated or have passed through AS 300.

Example

SW1's BGP table prior to route filtering.

```
SW1#show bgp ipv6
BGP table version is 2, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network             Next Hop               Metric LocPrf Weight Path
*> 2001:db8:200::/64
                    2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                                0             0 200 i
*> 2001:db8:300::/64
                    2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                                0             0 200 300 i
*> 2001:db8:400::/64
                    2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                                0             0 200 400 i
Number of prefixes 3
```

1. Create an AS path ACL on SW1 to permit prefixes that pass through AS 300.

- The prefix must transit AS 300 to be allowed installation in the BGP table.

```
SW1(config)#ip as-path access-list PERMIT_AS_LIST permit _300_
```

2. Create a route map to match on the AS path ACL.

```
SW1(config)#route-map PERMIT_AS_MAP permit 10
SW1(config-route-map)#match as-path PERMIT_AS_LIST
SW1(config-route-map)#exit
```

3. Assign the route map to the peering to apply the route filtering.

```
SW1(config)#router bgp 100
SW1(config-router)#address-family ipv6
SW1(config-router-af)#neighbor 2001:db8:10::2 route-map PERMIT_AS_MAP
   in
```

4. Clear the BGP session to the eBGP neighbor in AS 200 to apply the change.

```
SW1#clear bgp ipv6 2001:db8:10::2
```

Verify by:

- Viewing the BGP table. (Only the prefix originating from AS 300 is now present)
- Viewing the successful matches to the route map.
- Using the show bgp ipv6 neighbors command to see the accepted prefix counter and the route map applied to the incoming advertisements.

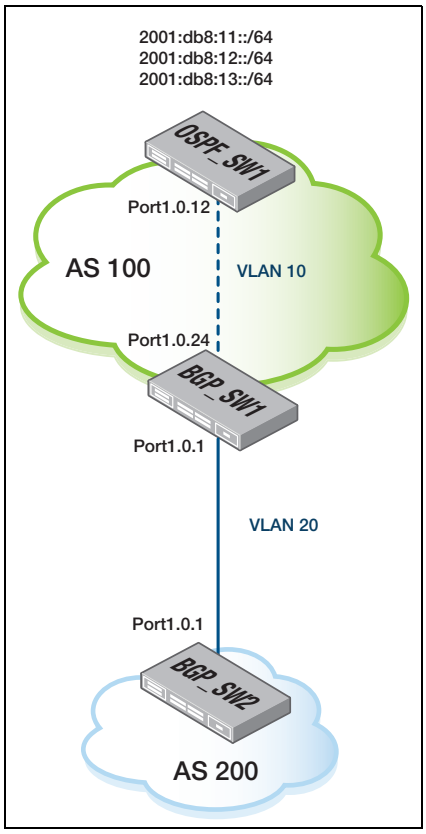
```
SW1#show bgp ipv6
BGP table version is 3, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network            Next Hop                    Metric LocPrf Weight Path
*> 2001:db8:300::/64
                   2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                                    0             0  200 300 i
Number of prefixes 1
```

```
SW1#show bgp ipv6 route-map PERMIT_AS_MAP
BGP table version is 3, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network            Next Hop                    Metric LocPrf Weight Path
*> 2001:db8:300::/64
                   2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                                    0             0 200 300 i
Number of prefixes 1
```

```
SW1#show bgp ipv6 neighbors
BGP neighbor is 2001:db8:10::2, remote AS 200, local AS 100, external link
  BGP version 4, remote router ID 2.2.2.2
  BGP state = Established, up for 00:00:9
  Last read 00:00:29, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
   Route refresh: advertised and received (old and new)
   4-Octet ASN Capability: advertised and received
   Address family IPv6 Unicast: advertised and received
 Received 18 messages, 0 notifications, 0 in queue
 Sent 9 messages, 1 notifications, 0 in queue
 Route refresh request: received 0, sent 0
 Minimum time between advertisement runs is 30 seconds
For address family: IPv6 Unicast
 BGP table version 3, neighbor version 3
 Index 1, Offset 0, Mask 0x2
 Community attribute sent to this neighbor (both)
 Inbound path policy configured
 Route map for incoming advertisements is *PERMIT_AS_MAP
 1 accepted prefixes
 0 announced prefixes

Connections established 2; dropped 1
Local host: 2001:db8:10::1, Local port: 41720
Foreign host: 2001:db8:10::2, Foreign port: 179
Nexthop: 1.1.1.1
Nexthop global: 2001:db8:10::1
Nexthop local: fe80::eecd:6dff:fe52:b789
BGP connection: shared network
Last Reset: 00:00:29, due to BGP Notification sent
Notification Error Message: (Cease/Other Configuration Change.)
```

Using distribute lists to filter routing updates - Inbound method

Scenario



Objective

- Filter prefixes 2001:db8:11::/64 and 2001:db8:13::/64 from updates received by BGP_SW2.
- Permit all other prefixes that may be advertised from BGP_SW1.

View of BGP_SW2's BGP table prior to filtering

Prefixes 2001:db8:11::/64 and 2001:db8:13::/64 are not yet filtered.

```
BGP_SW2#sh bgp ipv6
BGP table version is 13, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:11::/64 2001:db8:1::1           20             0 100 ?
*> 2001:db8:12::/64 2001:db8:1::1           20             0 100 ?
*> 2001:db8:13::/64 2001:db8:1::1           20             0 100 ?

Number of prefixes 3
```

1. Configure an ACL on BGP_SW2 denying only the two prefixes defined.

- The objective is to deny only the select two prefixes.
- Any other prefixes must be permitted.

```
BGP_SW2(config)#ipv6 access-list standard FILTER_IN deny
   2001:db8:11::/64
BGP_SW2(config)#ipv6 access-list standard FILTER_IN deny
   2001:db8:13::/64
BGP_SW2(config)#ipv6 access-list standard FILTER_IN permit any
```

2. Apply distribute list to filter the two prefixes from updates sent by neighbor 2001:db8:1::1 (BGP_SW1).

- Prefixes matching the deny statements in ACL "FILTER_IN" are not permitted to be installed in the BGP table.
- All other prefixes are allowed.

```
BGP_SW2(config)#router bgp 200
BGP_SW2(config-router)#address-family ipv6
BGP_SW2(config-router-af)#neighbor 2001:db8:1::1 distribute-list
   FILTER_IN in
```

3. Clear the BGP neighbor session to apply the filtering.

```
BGP_SW2#clear bgp ipv6 2001:db8:1::1
```

Verify by viewing the:

- BGP table.
- BGP neighborship (FILTER_IN list is applied).
- IPv6 access list.

```
BGP_SW2#show bgp ipv6
BGP table version is 26, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric   LocPrf     Weight   Path
*> 2001:db8:12::/64  2001:db8:1::1          20                    0    100   ?

Number of prefixes 1
```

```
BGP_SW2#show bgp ipv6 neighbors
BGP neighbor is 2001:db8:1::1, remote AS 100, local AS 200, external link
  BGP version 4, remote router ID 1.1.1.1
  BGP state = Established, up for 00:01:48
  Last read 00:01:48, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    4-Octet ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 63 messages, 0 notifications, 0 in queue
  Sent 58 messages, 1 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
  Update source is lo
 For address family: IPv6 Unicast
  BGP table version 26, neighbor version 26
  Index 1, Offset 0, Mask 0x2
  Community attribute sent to this neighbor (both)
  Inbound path policy configured
  Incoming update network filter list is *FILTER_IN
  1 accepted prefixes
  0 announced prefixes

 Connections established 2; dropped 1
  External BGP neighbor may be up to 2 hops away.
Local host: 2001:db8:2::1, Local port: 37042
Foreign host: 2001:db8:1::1, Foreign port: 179
Nexthop: 2.2.2.2
Nexthop global: 2001:db8:2::1
Nexthop local: ::
BGP connection: non shared network
Last Reset: 00:01:49, due to BGP Notification sent
Notification Error Message: (Cease/Other Configuration Change.)
```

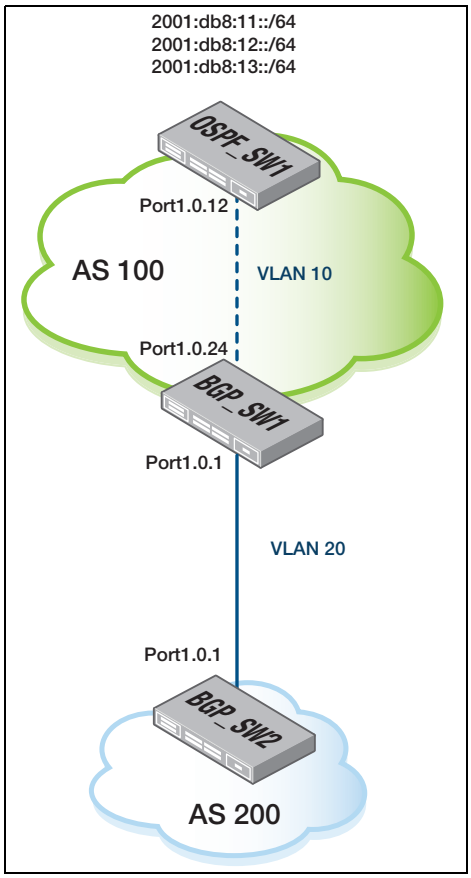
```
BGP_SW2#show ipv6 access-list standard FILTER_IN
Named Standard IPv6 access list FILTER_IN
   10 deny   2001:db8:11::/64
   20 deny   2001:db8:13::/64
   30 permit any
```

Configuration of BGP_SW2

```
BGP_SW2#
!
hostname BGP_SW2
!
ipv6 access-list standard FILTER_IN deny 2001:db8:11::/64
ipv6 access-list standard FILTER_IN deny 2001:db8:13::/64
ipv6 access-list standard FILTER_IN permit any
!
vlan database
 vlan 20,40 state enable
!
interface port1.0.1
 switchport access vlan 20
!
interface port1.0.2
 switchport access vlan 40
!
interface lo
 ipv6 address 2001:db8:2::1/64
!
interface vlan20
 ipv6 address 2001:db8:20::2/64
!
interface vlan40
 ipv6 address 2001:db8:40::1/64
!
router bgp 200
 bgp router-id 2.2.2.2
 no bgp default ipv4-unicast
 neighbor 2001:db8:1::1 remote-as 100
 neighbor 2001:db8:1::1 ebgp-multihop 2
 neighbor 2001:db8:1::1 update-source lo
 !
 address-family ipv6
 neighbor 2001:db8:1::1 activate
 neighbor 2001:db8:1::1 distribute-list FILTER_IN in
 exit-address-family
!
ipv6 route 2001:db8:1::1/128 fe80::200:cdff:fe2a:eaf1 vlan20
ipv6 route 2001:db8:3::1/128 fe80::200:cdff:fe2a:e635 vlan40
!
end
```

Using distribute lists to filter routing updates - Outbound method

Scenario



Objective

- Filter prefixes 2001:db8:11::/64 and 2001:db8:13::/64 being advertised by BGP_SW1 to BGP_SW2.
- Permit all other prefixes that may be advertised by BGP_SW1.

View of BGP_SW2's BGP table prior to filtering

Prefixes 2001:db8:11::/64 and 2001:db8:13::/64 are not yet filtered.

```
BGP_SW2#show bgp ipv6
BGP table version is 3, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:11::/64  2001:db8:1::1            20             0 100 ?
*> 2001:db8:12::/64  2001:db8:1::1            20             0 100 ?
*> 2001:db8:13::/64  2001:db8:1::1            20             0 100 ?

Number of prefixes 3
```

1. Configure ACL on BGP_SW1 denying only the two prefixes defined.

- Objective is to deny only the select two prefixes.
- Any other prefixes must be permitted.

```
BGP_SW1(config)#ipv6 access-list standard FILTER_OUT deny
   2001:db8:11::/64
BGP_SW1(config)#ipv6 access-list standard FILTER_OUT deny
   2001:db8:13::/64
BGP_SW1(config)#ipv6 access-list standard FILTER_OUT permit any
```

2. Apply distribute list to filter the two prefixes from updates sent to BGP neighbor BGP_SW2.

- Prefixes matching the deny statements in ACL "FILTER_OUT" are not permitted to advertised to BGP_SW2 in BGP Updates.
- All other prefixes are allowed.

```
BGP_SW1(config)#router bgp 100
BGP_SW1(config-router)#address-family ipv6
BGP_SW1(config-router-af)#neighbor 2001:db8:2::1 distribute-list
   FILTER_OUT out
BGP_SW1(config-router-af)#end
```

3. Clear the BGP neighbor session to apply the filtering.

```
BGP_SW1#clear bgp ipv6 2001:db8:2::1
```

Verify by viewing the:

- BGP table.
- BGP neighborship (FILTER_OUT list is applied).
- IPv6 access list.

```
BGP_SW2#show bgp ipv6
BGP table version is 9, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:12::/64  2001:db8:1::1           20             0 100 ?

Number of prefixes 1
```

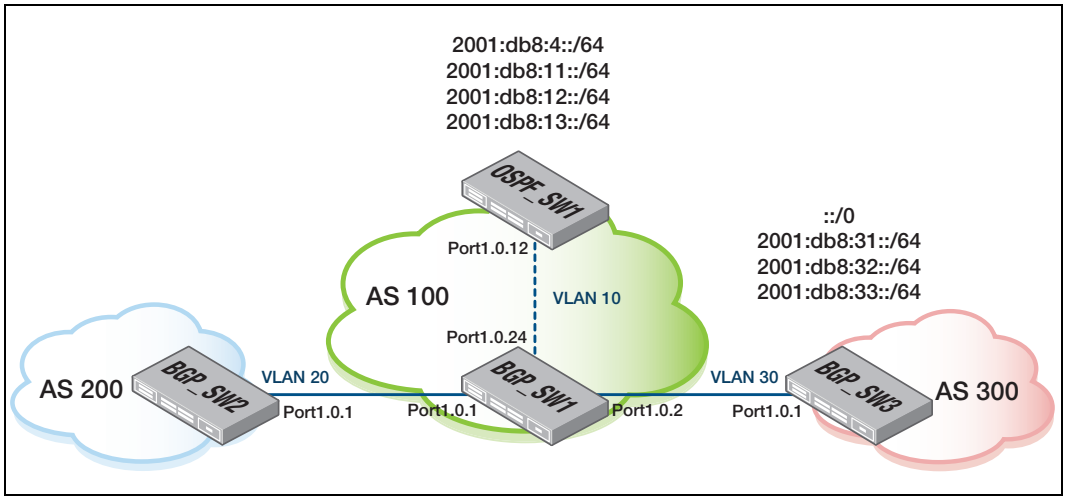
```
BGP_SW2#show bgp ipv6 neighbors
BGP neighbor is 2001:db8:1::1, remote AS 100, local AS 200, external link
  BGP version 4, remote router ID 1.1.1.1
  BGP state = Established, up for 00:01:36
  Last read 00:01:36, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    4-Octet ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 103 messages, 1 notifications, 0 in queue
  Sent 95 messages, 2 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
  Update source is lo
 For address family: IPv6 Unicast
  BGP table version 9, neighbor version 9
  Index 1, Offset 0, Mask 0x2
  Community attribute sent to this neighbor (both)
  1 accepted prefixes
  0 announced prefixes

 Connections established 4; dropped 3
  External BGP neighbor may be up to 2 hops away.
Local host: 2001:db8:2::1, Local port: 179
Foreign host: 2001:db8:1::1, Foreign port: 49590
Nexthop: 2.2.2.2
Nexthop global: 2001:db8:2::1
Nexthop local: ::
BGP connection: non shared network
Last Reset: 00:01:37, due to BGP Notification received
Notification Error Message: (Cease/Other Configuration Change.)
```

```
BGP_SW2#show ipv6 access-list
Named Standard IPv6 access list FILTER_IN
   10 deny   2001:db8:11::/64
   20 deny   2001:db8:13::/64
   30 permit any
```

Using BGP Outbound Route Filtering to dynamically filter outbound updates

Tip: Use Outbound Route Filtering (ORF) before establishing a neighbor adjacency to save bandwidth and processing.

Scenario



Objective

- Use ORF to dynamically filter outbound updates from BGP_SW1 in AS 100 to BGP_SW2 in AS 200.
- BGP_SW2 is the CE (Customer Edge) for AS 200.
- BGP_SW1 is the PE (Provider Edge) for AS 100. (Simulating an ISP for AS 200.)
- BGP_SW2 in AS 200, wants to receive only prefixes 2001:db8:4::/64 (from OSPF_SW1) and ::/0 from BGP_SW3.
- BGP_SW2 is to use ORF to tell BGP_SW1 what prefixes it wants to see, and filter the rest of the prefixes from BGP outbound updates to BGP_SW2.

View of BGP_SW2's BGP table prior to route filtering

```
BGP_SW2#show bgp ipv6
BGP table version is 18, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> ::/0             2001:db8:1::1            0             0 100 300 i
*> 2001:db8:4::/64  2001:db8:1::1           20             0 100 ?
*> 2001:db8:11::/64 2001:db8:1::1           20             0 100 ?
*> 2001:db8:12::/64 2001:db8:1::1           20             0 100 ?
*> 2001:db8:13::/64 2001:db8:1::1           20             0 100 ?
*> 2001:db8:31::/64 2001:db8:1::1            0             0 100 300 i
*> 2001:db8:32::/64 2001:db8:1::1            0             0 100 300 i
*> 2001:db8:33::/64 2001:db8:1::1            0             0 100 300 i

Number of prefixes 8
```

1. Create a prefix list on BGP_SW2 to select the prefixes desired in BGP updates from BGP_SW1 in AS 100.

```
BGP_SW2(config)#ipv6 prefix-list UPDATE_LIST seq 10 permit ::/0
BGP_SW2(config)#ipv6 prefix-list UPDATE_LIST seq 20 permit
   2001:db8:4::/64
```

2. On BGP_SW2 enable capability to send ORF to neighbor BGP_SW1 and apply the prefix list.

Prefixes defined in the prefix list "UPDATE_LIST" are to be sent to BGP_SW1 advising these are the only prefixes we want to see.

```
BGP_SW2(config)#router bgp 200
BGP_SW2(config-router)#address-family ipv6
BGP_SW2(config-router-af)#neighbor 2001:db8:1::1 capability orf prefix-
   list send
BGP_SW2(config-router-af)#neighbor 2001:db8:1::1 prefix-list UPDATE_LIST
   in
```

3. On BGP_SW1 enable capability to receive ORF from neighbor BGP_SW2.

- Enabling the ORF receive capability means prefixes requested from BGP_SW2 will be advertised if they are present in BGP_SW1's BGP table.
- Prefixes not present in the ORF list will be filtered from routing updates to BGP_SW2.

```
BGP_SW1(config)#router bgp 100
BGP_SW1(config-router)#address-family ipv6
BGP_SW1(config-router-af)#neighbor 2001:db8:2::1 capability orf prefix-
   list receive
```

Verify by viewing the:

- BGP table on BGP_SW2, (desired prefixes are present, others are not present).
- Number of prefixes received on BGP_SW2 in BGP updates from BGP_SW, (Prefixes received = 2).
- Neighbor output on BGP_SW2, with show bgp ipv6 neighbors | begin AF command.
- Neighbor output on BGP_SW1, with show bgp ipv6 neighbors | begin AF command.

Note: Clearing the BGP session is not required as when the ORF capability is configured, the session is reset.

```
BGP_SW2#show bgp ipv6
BGP table version is 27, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> ::/0             2001:db8:1::1            0             0 100 300 i
*> 2001:db8:4::/64  2001:db8:1::1           20             0 100 ?

Number of prefixes 2
```

```
BGP_SW2#show bgp ipv6 summary
BGP router identifier 2.2.2.2, local AS number 200
BGP table version is 27
2 BGP AS-PATH entries
0 BGP community entries

Neighbor        V         AS   MsgRc  MsgSnt TblVer InOutQ Up/Down State/PfxRcd
2001:db8:1::1   4         100     242     237     27 0/0    00:10:32        2

Number of neighbors 1
```

```
BGP_SW2#show bgp ipv6 neighbors | begin AF

...skipping
  AF-dependant capabilities:
    Outbound Route Filter (ORF) type (64) Prefix-list:
      Send-mode: advertised
      Receive-mode: received
    Outbound Route Filter (ORF) type (128) Prefix-list:
      Send-mode: advertised
      Receive-mode: received
  Outbound Route Filter (ORF): sent;
  Community attribute sent to this neighbor (both)
  Inbound path policy configured
  Incoming update prefix filter list is *UPDATE_LIST
  2 accepted prefixes
  0 announced prefixes

 Connections established 3; dropped 2
  External BGP neighbor may be up to 2 hops away.
Local host: 2001:db8:2::1, Local port: 179
Foreign host: 2001:db8:1::1, Foreign port: 53546
Nexthop: 2.2.2.2
Nexthop global: 2001:db8:2::1
Nexthop local: ::
BGP connection: non shared network
Last Reset: 00:12:56, due to BGP Notification received
Notification Error Message: (Cease/Other Configuration Change.)
```

```
BGP_SW1#show bgp ipv6 neighbors 2001:db8:2::1 | begin AF

...skipping
  AF-dependant capabilities:
    Outbound Route Filter (ORF) type (64) Prefix-list:
      Send-mode: received
      Receive-mode: advertised
    Outbound Route Filter (ORF) type (128) Prefix-list:
      Send-mode: received
      Receive-mode: advertised
  Outbound Route Filter (ORF): received (2 entries)
  Community attribute sent to this neighbor (both)
  0 accepted prefixes
  2 announced prefixes

 Connections established 3; dropped 2
  External BGP neighbor may be up to 2 hops away.
Local host: 2001:db8:1::1, Local port: 53546
Foreign host: 2001:db8:2::1, Foreign port: 179
Nexthop: 1.1.1.1
Nexthop global: 2001:db8:1::1
Nexthop local: ::
BGP connection: non shared network
Last Reset: 00:14:10, due to BGP Notification sent
Notification Error Message: (Cease/Other Configuration Change.)
```

BGP_SW1 (PE) configuration

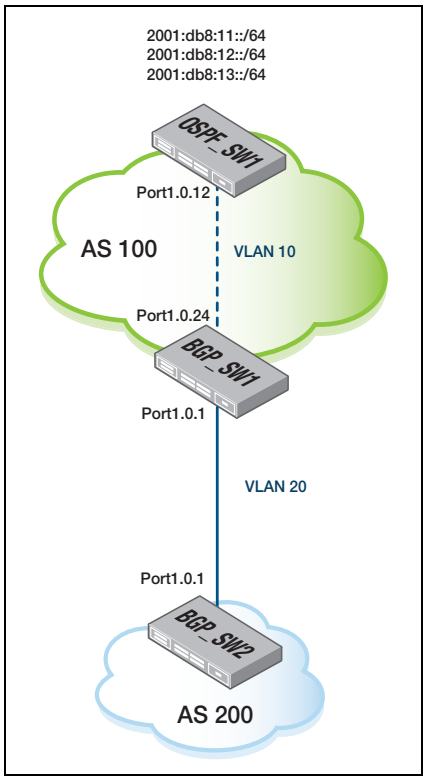
```
BGP_SW1#
!
hostname BGP_SW1
!
vlan database
 vlan 10,20,30 state enable
!
interface port1.0.1
 switchport access vlan 20
!
interface port1.0.2
 switchport access vlan 30
!
interface port1.0.24
 switchport access vlan 10
!
interface lo
 ipv6 address 2001:db8:1::1/64
!
interface vlan10
 ipv6 address 2001:db8:10::1/64
 ipv6 router ospf area 0
!
interface vlan20
 ipv6 address 2001:db8:20::1/64
!
interface vlan30
 ipv6 address 2001:db8:30::1/64
 ipv6 enable
!
ipv6 ospf display route single-line
!
router ipv6 ospf
 router-id 1.1.1.1
!
router bgp 100
 bgp router-id 1.1.1.1
 no bgp default ipv4-unicast
 neighbor 2001:db8:2::1 remote-as 200
 neighbor 2001:db8:2::1 ebgp-multihop 2
 neighbor 2001:db8:2::1 update-source lo
 neighbor 2001:db8:3::1 remote-as 300
 neighbor 2001:db8:3::1 ebgp-multihop 2
 neighbor 2001:db8:3::1 update-source lo
 !
 address-family ipv6
 redistribute ospf
 neighbor 2001:db8:2::1 activate
 neighbor 2001:db8:2::1 capability orf prefix-list receive
 neighbor 2001:db8:3::1 activate
 exit-address-family
!
ipv6 route 2001:db8:2::1/128 fe80::209:41ff:fefb:c323 vlan20
ipv6 route 2001:db8:3::1/128 fe80::200:cdff:fe2a:e635 vlan30
!
end
```

BGP_SW2 (CE) configuration

```
BGP_SW2#
!
hostname BGP_SW2
!
vlan database
 vlan 20,40 state enable
!
interface port1.0.1
 switchport access vlan 20
!
interface port1.0.2
 switchport access vlan 40
!
interface lo
 ipv6 address 2001:db8:2::1/64
!
interface vlan20
 ipv6 address 2001:db8:20::2/64
!
interface vlan40
 ipv6 address 2001:db8:40::1/64
!
router bgp 200
 bgp router-id 2.2.2.2
 no bgp default ipv4-unicast
 neighbor 2001:db8:1::1 remote-as 100
 neighbor 2001:db8:1::1 ebgp-multihop 2
 neighbor 2001:db8:1::1 update-source lo
 !
 address-family ipv6
 neighbor 2001:db8:1::1 activate
 neighbor 2001:db8:1::1 capability orf prefix-list send
 neighbor 2001:db8:1::1 prefix-list UPDATE_LIST in
 exit-address-family
!
ipv6 route 2001:db8:1::1/128 fe80::200:cdff:fe2a:eaf1 vlan20
ipv6 route 2001:db8:3::1/128 fe80::200:cdff:fe2a:e635 vlan40
!
ipv6 prefix-list UPDATE_LIST seq 10 permit ::/0
ipv6 prefix-list UPDATE_LIST seq 20 permit 2001:db8:4::/64
!
end
```

Using Prefix lists to filter routing updates - Inbound method

Scenario



Objective

- Filter prefixes 2001:db8:11::/64 and 2001:db8:13::/64 being advertised by BGP_SW1 to BGP_SW2.
- Permit all other prefixes that may be advertised by BGP_SW1.

View of BGP_SW2's BGP table prior to filtering:

```
BGP_SW2#show bgp ipv6
BGP table version is 2, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:11::/64 2001:db8:1::1            0             0 100 i
*> 2001:db8:12::/64 2001:db8:1::1            0             0 100 i
*> 2001:db8:13::/64 2001:db8:1::1            0             0 100 i
```

1. Configure a Prefix List on BGP_SW2 denying only the two prefixes defined.

```
BGP_SW2(config)#ipv6 prefix-list FILTER_IN seq 10 deny 2001:db8:11::/64
BGP_SW2(config)#ipv6 prefix-list FILTER_IN seq 20 deny 2001:db8:13::/64
BGP_SW2(config)#ipv6 prefix-list FILTER_IN seq 30 permit any
```

2. Apply the Prefix List to filter the two prefixes from updates sent to BGP_SW2.

```
BGP_SW2(config)#router bgp 200
BGP_SW2(config-router)#address-family ipv6
BGP_SW2(config-router-af)#neighbor 2001:db8:1::1 prefix-list FILTER_IN in
```

3. Clear the BGP neighbor session to apply the filtering.

```
BGP_SW2#clear bgp ipv6 2001:db8:1::1
```

Verify by viewing the:

- BGP table on BGP_SW2.
- BGP neighborship on BGP_SW2, (FILTER_IN list is applied).
- IPv6 prefix list on BGP_SW2.

```
BGP_SW2#show bgp ipv6
BGP table version is 6, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:12::/64 2001:db8:1::1            0             0 100 i

Number of prefixes 1
```

```
BGP_SW2#show bgp ipv6 neighbors 2001:db8:1::1
BGP neighbor is 2001:db8:1::1, remote AS 100, local AS 200, external link
  BGP version 4, remote router ID 1.1.1.1
  BGP state = Established, up for 00:00:31
  Last read 00:00:31, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    4-Octet ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 539 messages, 1 notifications, 0 in queue
  Sent 538 messages, 2 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
  Update source is lo
 For address family: IPv6 Unicast
  BGP table version 6, neighbor version 6
  Index 1, Offset 0, Mask 0x2
  Community attribute sent to this neighbor (both)
  Inbound path policy configured
  Incoming update prefix filter list is *FILTER_IN
  1 accepted prefixes
  0 announced prefixes

Connections established 4; dropped 3
  External BGP neighbor may be up to 2 hops away.
Local host: 2001:db8:2::1, Local port: 55304
Foreign host: 2001:db8:1::1, Foreign port: 179
Nexthop: 2.2.2.2
Nexthop global: 2001:db8:2::1
Nexthop local: ::
BGP connection: non shared network
Last Reset: 00:00:32, due to BGP Notification sent
Notification Error Message: (Cease/Other Configuration Change.)
```

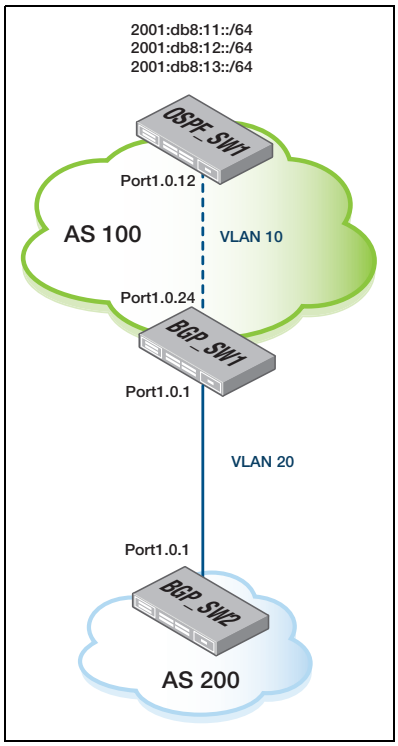
```
BGP_SW2#show ipv6 prefix-list detail
Prefix-list with the last deletion/insertion: FILTER_IN
ipv6 prefix-list FILTER_IN:
   count: 3, range entries: 0, sequences: 10 - 30
    ripd:
     seq 10 deny 2001:db8:11::/64 (hit count: 0, refcount: 0)
     seq 20 deny 2001:db8:13::/64 (hit count: 0, refcount: 0)
     seq 30 permit any (hit count: 0, refcount: 0)
    ripngd:
     seq 10 deny 2001:db8:11::/64 (hit count: 0, refcount: 0)
     seq 20 deny 2001:db8:13::/64 (hit count: 0, refcount: 0)
     seq 30 permit any (hit count: 0, refcount: 0)
    ospfd:
     seq 10 deny 2001:db8:11::/64 (hit count: 0, refcount: 0)
     seq 20 deny 2001:db8:13::/64 (hit count: 0, refcount: 0)
     seq 30 permit any (hit count: 0, refcount: 0)
    ospf6d:
     seq 10 deny 2001:db8:11::/64 (hit count: 0, refcount: 0)
     seq 20 deny 2001:db8:13::/64 (hit count: 0, refcount: 0)
     seq 30 permit any (hit count: 0, refcount: 0)
    bgpd:
     seq 10 deny 2001:db8:11::/64 (hit count: 1, refcount: 3)
     seq 20 deny 2001:db8:13::/64 (hit count: 1, refcount: 2)
     seq 30 permit any (hit count: 1, refcount: 1)
```

BGP_SW2 Configuration

```
BGP_SW2#
!
hostname BGP_SW2
!
vlan database
 vlan 20,40 state enable
!
interface port1.0.1
 switchport access vlan 20
!
interface port1.0.2
 switchport access vlan 40
!
interface lo
 ipv6 address 2001:db8:2::1/64
!
interface vlan20
 ipv6 address 2001:db8:20::2/64
!
interface vlan40
 ipv6 address 2001:db8:40::1/64
!
router bgp 200
 bgp router-id 2.2.2.2
 no bgp default ipv4-unicast
 neighbor 2001:db8:1::1 remote-as 100
 neighbor 2001:db8:1::1 ebgp-multihop 2
 neighbor 2001:db8:1::1 update-source lo
 !
 address-family ipv6
 neighbor 2001:db8:1::1 activate
 neighbor 2001:db8:1::1 prefix-list FILTER_IN in
 exit-address-family
!
ipv6 route 2001:db8:1::1/128 fe80::200:cdff:fe2a:eaf1 vlan20
ipv6 route 2001:db8:3::1/128 fe80::200:cdff:fe2a:e635 vlan40
!
ipv6 prefix-list FILTER_IN seq 10 deny 2001:db8:11::/64
ipv6 prefix-list FILTER_IN seq 20 deny 2001:db8:13::/64
ipv6 prefix-list FILTER_IN seq 30 permit any
!
end
```

Using Prefix lists to filter routing updates - Outbound method

Scenario



Objective

- Filter prefixes 2001:db8:11::/64 and 2001:db8:13::/64 being advertised by BGP_SW1 to BGP_SW2.
- Permit all other prefixes that may be advertised by BGP_SW1.

View of BGP_SW2's BGP table prior to filtering:

```
BGP_SW2#show bgp ipv6
BGP table version is 12, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:11::/64 2001:db8:1::1            0             0 100 i
*> 2001:db8:12::/64 2001:db8:1::1            0             0 100 i
*> 2001:db8:13::/64 2001:db8:1::1            0             0 100 i

Number of prefixes 3
```

1. Configure a Prefix List on BGP_SW1 denying only the two prefixes defined.

```
BGP_SW1(config)#ipv6 prefix-list FILTER_OUT seq 10 deny 2001:db8:11::/64
BGP_SW1(config)#ipv6 prefix-list FILTER_OUT seq 20 deny 2001:db8:13::/64
BGP_SW1(config)#ipv6 prefix-list FILTER_OUT seq 30 permit any
```

2. Apply the Prefix List to filter the two prefixes from updates sent to BGP neighbor BGP_SW2.

```
BGP_SW1(config)#router bgp 100
BGP_SW1(config-router)#address-family ipv6
BGP_SW1(config-router-af)#neighbor 2001:db8:2::1 prefix-list FILTER_OUT
   out
```

3. Clear the BGP neighbor session to apply the filtering.

```
BGP_SW1#clear bgp ipv6 2001:db8:2::1
```

Verify by viewing the:

- BGP table on BGP_SW2.
- BGP neighborship on BGP_SW1 (FILTER_OUT list is applied).
- IPv6 prefix list on BGP_SW1.

```
BGP_SW2#show bgp ipv6
BGP table version is 20, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight     Path
*> 2001:db8:12::/64 2001:db8:1::1            0             0     100    i

Number of prefixes 1
```

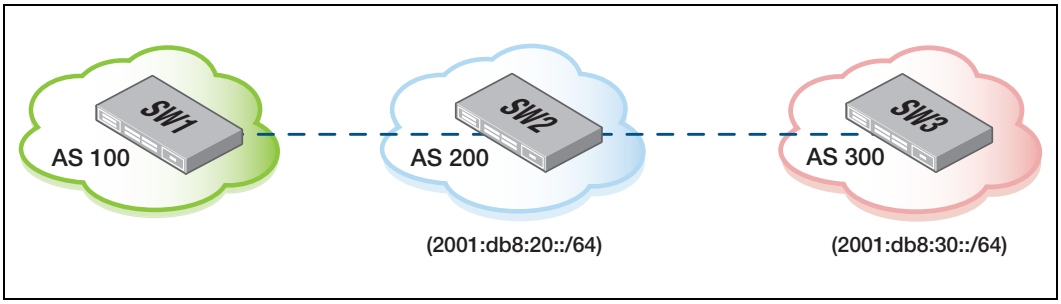
```
BGP_SW1#show bgp ipv6 neighbors 2001:db8:2::1
BGP neighbor is 2001:db8:2::1, remote AS 200, local AS 100, external link
  BGP version 4, remote router ID 2.2.2.2
  BGP state = Established, up for 00:01:10
  Last read 00:01:10, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    4-Octet ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 476 messages, 0 notifications, 0 in queue
  Sent 475 messages, 2 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
  Update source is lo
 For address family: IPv6 Unicast
  BGP table version 16, neighbor version 16
  Index 1, Offset 0, Mask 0x2
  Community attribute sent to this neighbor (both)
  Outbound path policy configured
  Outgoing update prefix filter list is *FILTER_OUT
  0 accepted prefixes
  1 announced prefixes

 Connections established 2; dropped 1
  External BGP neighbor may be up to 2 hops away.
Local host: 2001:db8:1::1, Local port: 179
Foreign host: 2001:db8:2::1, Foreign port: 56385
Nexthop: 1.1.1.1
Nexthop global: 2001:db8:1::1
Nexthop local: ::
BGP connection: non shared network
Last Reset: 00:01:10, due to BGP Notification sent
Notification Error Message: (Cease/Administratively Reset.)
```

```
BGP_SW1#show ipv6 prefix-list detail
Prefix-list with the last deletion/insertion: FILTER_OUT
ipv6 prefix-list FILTER_OUT:
   count: 3, range entries: 0, sequences: 10 - 30
    ripd:
     seq 10 deny 2001:db8:11::/64 (hit count: 0, refcount: 0)
     seq 20 deny 2001:db8:13::/64 (hit count: 0, refcount: 0)
     seq 30 permit any (hit count: 0, refcount: 0)
    ripngd:
     seq 10 deny 2001:db8:11::/64 (hit count: 0, refcount: 0)
     seq 20 deny 2001:db8:13::/64 (hit count: 0, refcount: 0)
     seq 30 permit any (hit count: 0, refcount: 0)
    ospfd:
     seq 10 deny 2001:db8:11::/64 (hit count: 0, refcount: 0)
     seq 20 deny 2001:db8:13::/64 (hit count: 0, refcount: 0)
     seq 30 permit any (hit count: 0, refcount: 0)
    ospf6d:
     seq 10 deny 2001:db8:11::/64 (hit count: 0, refcount: 0)
     seq 20 deny 2001:db8:13::/64 (hit count: 0, refcount: 0)
     seq 30 permit any (hit count: 0, refcount: 0)
    bgpd:
     seq 10 deny 2001:db8:11::/64 (hit count: 1, refcount: 3)
     seq 20 deny 2001:db8:13::/64 (hit count: 1, refcount: 2)
     seq 30 permit any (hit count: 1, refcount: 1)
```

BGP_SW1 configuration

```
BGP_SW1#
!
hostname BGP_SW1
!
vlan database
 vlan 10,20,30 state enable
!
interface port1.0.1
 switchport access vlan 20
!
interface port1.0.2
 switchport access vlan 30
!
interface port1.0.24
 switchport access vlan 10
!
interface lo
 ipv6 address 2001:db8:1::1/64
!
interface vlan10
 ipv6 address 2001:db8:10::1/64
 ipv6 router ospf area 0
!
interface vlan20
 ipv6 address 2001:db8:20::1/64
!
interface vlan30
 ipv6 address 2001:db8:30::1/64
 ipv6 enable
!
ipv6 ospf display route single-line
!
router bgp 100
 bgp router-id 1.1.1.1
 no bgp default ipv4-unicast
 neighbor 2001:db8:2::1 remote-as 200
 neighbor 2001:db8:2::1 ebgp-multihop 2
 neighbor 2001:db8:2::1 update-source lo
 neighbor 2001:db8:10::2 remote-as 10
 !
 address-family ipv6
 network 2001:db8:11::/64
 network 2001:db8:12::/64
 network 2001:db8:13::/64
 neighbor 2001:db8:2::1 activate
 neighbor 2001:db8:2::1 prefix-list FILTER_OUT out
 neighbor 2001:db8:10::2 activate
 exit-address-family
!
ipv6 route 2001:db8:2::1/128 fe80::209:41ff:fefb:c323 vlan20
ipv6 route 2001:db8:3::1/128 fe80::200:cdff:fe2a:e635 vlan30
!
ipv6 prefix-list FILTER_OUT seq 10 deny 2001:db8:11::/64
ipv6 prefix-list FILTER_OUT seq 20 deny 2001:db8:13::/64
ipv6 prefix-list FILTER_OUT seq 30 permit any
!
end
```

Using the BGP well-known community attribute (no-advertise) to perform route filtering



Use

- The no-advertise community is one of the predefined, well-known communities.
- The no-advertise community attribute informs the BGP peer not to advertise the route(s) to any peer (IBGP or EBGP peer).

Implementation example

A business which wants to only advertise a prefix to a single AS, and does not want the information propagated to any other AS.

Objective

- Use the no-advertise community attribute to prevent the 2001:db8:30::/64 prefix being advertised by SW2 in AS 200 to SW1 in AS 100.

View of SW1's BGP table prior to the no-advertise community being advertised:

```
SW1#show bgp ipv6
BGP table version is 4, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network          Next Hop                  Metric LocPrf Weight Path
*> 2001:db8:20::/64
                 2001:db8:1::2(fe80::eecd:6dff:fe20:c26b)
                                                0             0 200 i
*> 2001:db8:30::/64
                 2001:db8:1::2(fe80::eecd:6dff:fe20:c26b)
                                                0             0 200 300 i
Number of prefixes 2
```

1. Create an Access List to classify the prefix to have the community value applied.

```
SW3(config)#ipv6 access-list standard NO_ADVERTISE_LIST permit
   2001:db8:30::/64
```

2. Create a route map to match on the ACL and set the well-known 'no-advertise' community value.

- Sequence 10 will set the no-advertise community on a successful match of the IPv6 ACL.
- Sequence 20 will permit all other routes.

```
SW3(config)#route-map SET_COMMUNITY permit 10
SW3(config-route-map)#match ipv6 address NO_ADVERTISE_LIST
SW3(config-route-map)#set community no-advertise
SW3(config-route-map)#route-map SET_COMMUNITY permit 20
```

3. Apply the route map to the external peer.

```
SW3(config)#router bgp 300
SW3(config-router)#address-family ipv6
SW3(config-router-af)#neighbor 2001:db8:2::1 route-map SET_COMMUNITY
   out
```

4. Clear the session to apply the change.

```
SW3#clear bgp ipv6 2001:db8:2::1
```

Verify by viewing:

- SW1's BGP table now missing the prefix 2001:db8:30::/64. Prefix matched by the route map is no longer advertised to AS 100.
- The routes that match the no-advertise community on SW2.
- The specific prefix learned on SW2 from SW3. Community attribute is present.
- SW3's neighbor information to 2001:db8:2::1.

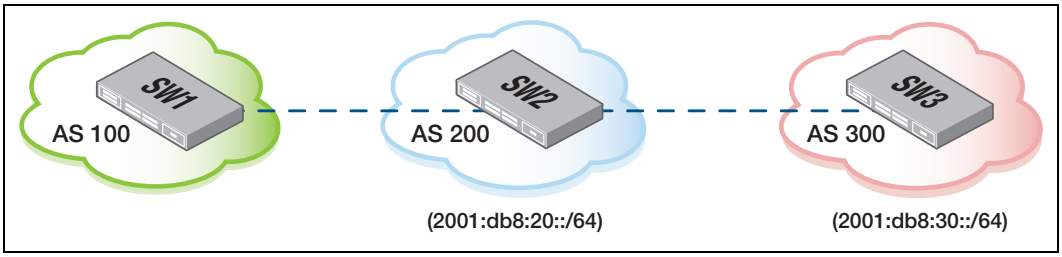
```
SW1#show bgp ipv6
BGP table version is 19, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network          Next Hop                            Metric LocPrf Weight Path
*> 2001:db8:20::/64
                 2001:db8:1::2(fe80::eecd:6dff:fe20:c26b)
                                                          0             0 200 i
Number of prefixes 1
```

```
SW2#show bgp ipv6 community
BGP table version is 4, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network            Next Hop                        Metric LocPrf Weight Path
*> 2001:db8:30::/64
                   2001:db8:2::2(fe80::eecd:6dff:fe20:c23e)
                                                        0             0 300 i
Number of prefixes 1
```

```
SW2#show bgp ipv6 2001:db8:30::/64
BGP routing table entry for 2001:db8:30::/64
Paths: (1 available, best #1, table Default-IP-Routing-Table, not advertised to
any peer)
 Not advertised to any peer
 300
   2001:db8:2::2(fe80::eecd:6dff:fe20:c23e) from 2001:db8:2::2 (3.3.3.3)
  (fe80::eecd:6dff:fe20:c23e)
    Origin IGP metric 0, localpref 100, valid, external, best
    Community: no-advertise
    Last update: Thu Jul 4 21:40:38 2013
```

```
SW3#show bgp ipv6 neighbors 2001:db8:2::1
BGP neighbor is 2001:db8:2::1, remote AS 200, local AS 300, external link
 BGP version 4, remote router ID 2.2.2.2
 BGP state = Established, up for 00:20:55
 Last read 00:20:55, hold time is 90, keepalive interval is 30 seconds
 Neighbor capabilities:
  Route refresh: advertised and received (old and new)
  4-Octet ASN Capability: advertised and received
  Address family IPv6 Unicast: advertised and received
 Received 51 messages, 0 notifications, 0 in queue
 Sent 52 messages, 0 notifications, 0 in queue
 Route refresh request: received 0, sent 0
 Minimum time between advertisement runs is 30 seconds
 For address family: IPv4 Unicast
  BGP table version 1, neighbor version 1
  Index 1, Offset 0, Mask 0x2
  Community attribute sent to this neighbor (both)
  0 accepted prefixes
  0 announced prefixes
For address family: IPv6 Unicast
 BGP table version 22, neighbor version 22
 Index 1, Offset 0, Mask 0x2
 Community attribute sent to this neighbor (both)
 Outbound path policy configured
 Route map for outgoing advertisements is *SET_COMMUNITY
 1 accepted prefixes
 1 announced prefixes

 Connections established 1; dropped 0
Local host: 2001:db8:2::2, Local port: 179
Foreign host: 2001:db8:2::1, Foreign port: 60209
Nexthop: 3.3.3.3
Nexthop global: 2001:db8:2::2
Nexthop local: fe80::eecd:6dff:fe20:c23e
BGP connection: shared network
```

Using the BGP well-known community attribute (no-export) to perform route filtering



Use

- The no-export community is one of the predefined, well-known communities.
- This well-known community informs the BGP peer not to advertise the route(s) to EBGP peers. The route must be kept within the AS.

Implementation example

A business which wants to only advertise a prefix to a single AS, and does not want the information propagated to any other AS.

Objective

Use the no-export community value to prevent the 2001:db8:30::/64 prefix being advertised by SW2 in AS 200 to SW1 in AS 100.

View of SW1's BGP table prior to the no-export community being advertised:

```
SW1#show bgp ipv6
BGP table version is 9, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network             Next Hop                    Metric LocPrf Weight Path
*> 2001:db8:20::/64
                    2001:db8:1::2(fe80::eecd:6dff:fe20:c26b)
                                                     0             0  20 i
*> 2001:db8:30::/64
                    2001:db8:1::2(fe80::eecd:6dff:fe20:c26b)
                                                     0             0 200 300 i
Number of prefixes 2
```

1. Create an Access List to classify the prefix to have the community value applied.

```
SW3(config)#ipv6 access-list standard NO_EXPORT_LIST permit
   2001:db8:30::/64
```

2. Create a Route map to match on the ACL and set the well-known 'no-export' community value.

- Sequence 10 will set the no-export community on a successful match of the IPv6 ACL.
- Sequence 20 will permit all other routes.

```
SW3(config)#route-map SET_COMMUNITY permit 10
SW3(config-route-map)#match ipv6 address NO_EXPORT_LIST
SW3(config-route-map)#set community no-export
SW3(config-route-map)#route-map SET_COMMUNITY permit 20
```

3. Apply the route map to the external peer.

```
SW3(config)#router bgp 300
SW3(config-router)#address-family ipv6
SW3(config-router-af)#neighbor 2001:db8:2::1 route-map SET_COMMUNITY
   out
```

4. Clear the session to apply the change.

```
SW3#clear bgp ipv6 2001:db8:2::1
```

Verify by viewing:

- SW1's BGP table, (prefix matched by the route map is no longer advertised to AS 100).
- The routes that match the no-export community on SW2.
- The specific prefix learned on SW2 from SW3 (community attribute is present).
- SW3's neighbor information to 2001:db8:2::1.

```
SW1#show bgp ipv6
BGP table version is 34, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network          Next Hop                            Metric LocPrf Weight Path
*> 2001:db8:20::/64
                 2001:db8:1::2(fe80::eecd:6dff:fe20:c26b)
                                                          0             0 200 i
Number of prefixes 1
```

```
SW2#show bgp ipv6 community
BGP table version is 128, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network          Next Hop                          Metric LocPrf Weight Path
*> 2001:db8:30::/64
                 2001:db8:2::2(fe80::eecd:6dff:fe20:c23e)
                                                        0             0 300 i
Number of prefixes 1
```

```
SW2#show bgp ipv6 2001:db8:30::/64
BGP routing table entry for 2001:db8:30::/64
Paths: (1 available, best #1, table Default-IP-Routing-Table, not advertised to
EBGP peer)
  Not advertised to any peer
  300
    2001:db8:2::2(fe80::eecd:6dff:fe20:c23e) from 2001:db8:2::2 (3.3.3.3)
    (fe80::eecd:6dff:fe20:c23e)
     Origin IGP metric 0, localpref 100, valid, external, best
     Community: no-export
     Last update: Thu Jul 4 22:06:56 2013
```

```
SW3#show bgp ipv6 neighbors 2001:db8:2::1
BGP neighbor is 2001:db8:2::1, remote AS 200, local AS 300, external link
  BGP version 4, remote router ID 2.2.2.2
  BGP state = Established, up for 00:03:51
  Last read 00:03:51, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    4-Octet ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 11 messages, 0 notifications, 0 in queue
  Sent 11 messages, 0 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
For address family: IPv6 Unicast
 BGP table version 5, neighbor version 4
 Index 1, Offset 0, Mask 0x2
 Community attribute sent to this neighbor (both)
 Outbound path policy configured
 Route map for outgoing advertisements is *SET_COMMUNITY
 1 accepted prefixes
 1 announced prefixes

 Connections established 1; dropped 0
Local host: 2001:db8:2::2, Local port: 179
Foreign host: 2001:db8:2::1, Foreign port: 36601
Nexthop: 3.3.3.3
Nexthop global: 2001:db8:2::2
Nexthop local: fe80::eecd:6dff:fe20:c23e
BGP connection: shared network
```

This chapter covers the following topics:

- BGP route aggregation rules
- Setup basic route aggregation in BGP4+ without suppressing longer prefixes in routing updates
- Setup basic route aggregation in BGP4+ and suppress more specific prefixes from being advertised
- Setup basic route aggregation to aggregate routes received from a neighboring AS
- Setup an unsuppress map to advertise longer prefixes suppressed by an aggregate address

CHAPTER 5

Route Aggregation

Route aggregation is used to:

- Suppress specific prefixes and advertise a single summary route.
- Decrease the number of prefixes being advertised and reduce routing table size.

BGP route aggregation rules

RFC4271 states:
"*Path Attributes that have different type codes cannot be aggregated together.*"

This means that the paths we want aggregated, must all have the same type of attributes.

Notable rules regarding MED (metric), ORIGIN and NEXT_HOP PA values for route aggregation

- MULTI_EXIT_DISC route aggregation rule:
  "*Routes that have different MULTI_EXIT_DISC attributes SHALL NOT be aggregated.*"

This means that if different BGP paths contain different MED values, the routes will not be aggregated.

Note: If multiple route sources require aggregation, (e.g. IGP and connected or static routes), use a route map to set the metric to zero when redistributing the routes into BGP.

- ORIGIN route aggregation rule:
  "*If at least one route among routes that are aggregated has ORIGIN with the value INCOMPLETE, then the aggregated route MUST have the ORIGIN attribute with the value INCOMPLETE. Otherwise, if at least one route among routes that are aggregated has ORIGIN with the value EGP, then the aggregated route MUST have the ORIGIN attribute with the value EGP. In all other cases, the value of the ORIGIN attribute of the aggregated route is IGP.*"

ORIGIN attribute of the aggregated route is set to the highest (least preferred) ORIGIN value out of all the paths being aggregated. (E.g. set to INCOMPLETE if any route aggregated has INCOMPLETE value, if no INCOMPLETE ORIGIN, then set to EGP if there are any of EGP ORIGIN. Otherwise set to IGP.)

- NEXT_HOP Attribute: By default, the AlliedWare Plus BGP implementation will aggregate longer prefixes which have different Next-Hop addresses. To change this behavior, configure bgp aggregate-nexthop-check under the global configuration mode.

To revert back to the default behaviour, in **global configuration mode**, use the command:

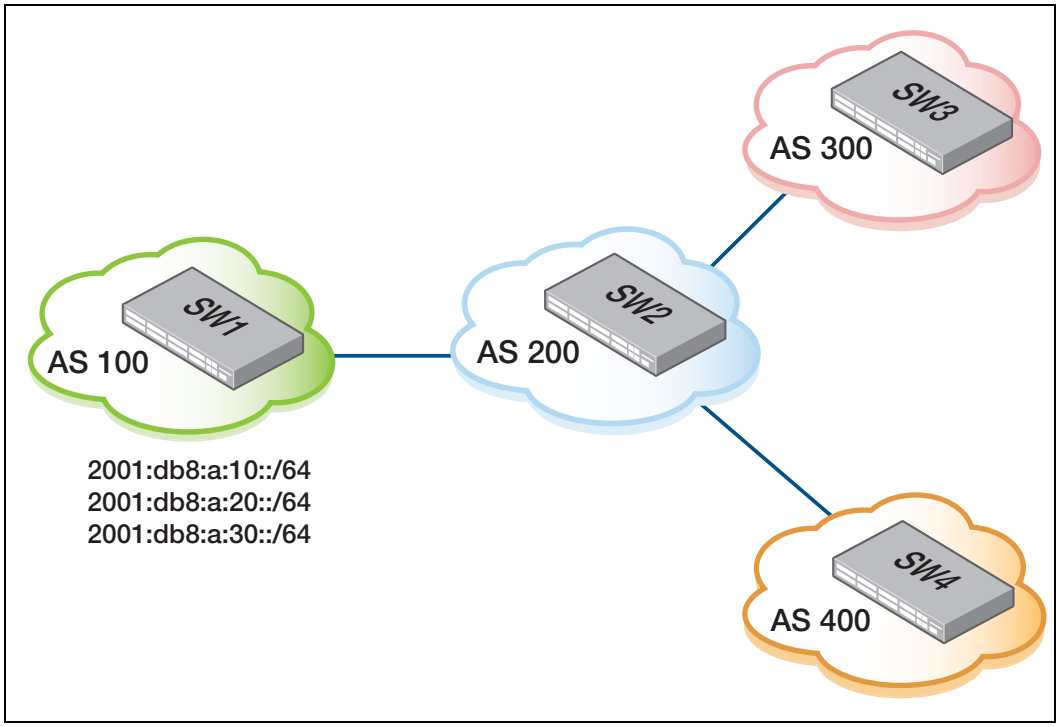
no bgp aggregate-nexthop-check

In summary:

- Paths which have different attribute type codes cannot be aggregated.
- Paths which have the same attribute type codes can be aggregated.
- Each route must have identical MED attributes to be aggregated.
- NEXT_HOP must be identical if the bgp aggregate-nexthop-check command is configured.
- ORIGIN attribute of aggregate is set to the least preferred ORIGIN value of the routes being aggregated.

For more information about route aggregation rules see:
http://tools.ietf.org/html/rfc4271#section-9.2.2.2

Setup basic route aggregation in BGP4+ without suppressing longer prefixes in routing updates



Route aggregation is used to:

- Aggregate longer prefixes and advertise a summary route.
- Decrease number of prefixes being advertised and reduce routing table size.

Objective

- Use the BGP4+ aggregate-address command under the IPv6 address family to advertise a summary route from SW1.
- More specific prefixes are NOT suppressed in this configuration.

SW2, SW3, and SW4's BGP tables prior to route aggregation

```
SW2#show bgp ipv6
BGP table version is 23, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network             Next Hop             Metric LocPrf Weight Path
*> 2001:db8:a:10::/64
                    2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                                 0             0  100 i
*> 2001:db8:a:20::/64
                    2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                                 0             0  100 i
*> 2001:db8:a:30::/64
                    2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                                 0             0  100 i
Number of prefixes 3
```

```
SW3#show bgp ipv6
BGP table version is 2, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network              Next Hop             Metric LocPrf Weight Path
*> 2001:db8:a:10::/64
                    2001:db8:20::1(fe80::eecd:6dff:fe20:c26b)
                                                 0             0  200 100  i
*> 2001:db8:a:20::/64
                    2001:db8:20::1(fe80::eecd:6dff:fe20:c26b)
                                                 0             0  200 100  i
*> 2001:db8:a:30::/64
                    2001:db8:20::1(fe80::eecd:6dff:fe20:c26b)
                                                 0             0  200 100  i
Number of prefixes 3
```

```
SW4#show bgp ipv6
BGP table version is 2, local router ID is 4.4.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network             Next Hop             Metric LocPrf Weight Path
*> 2001:db8:a:10::/64
                    2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                                 0             0  200 100 i
*> 2001:db8:a:20::/64
                    2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                                 0             0  200 100 i
*> 2001:db8:a:30::/64
                    2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                                 0             0  200 100 i
Number of prefixes 3
```

1. **Calculate the summary address with an appropriate subnet mask that matches the subnets to be summarized into a single prefix.**

- List the prefixes to summarize:

  `2001:0db8:000a:0010::/64`

  `2001:0db8:000a:0020::/64`

  `2001:0db8:000a:0030::/64`

- Each digit in an IPv6 address is 4 bits. Each colon-separated segment of the IPv6 address is 16 bits.
- The first three identical segments of each prefix is `2001:0db8:000a`, that is **48** bits to be used as part of the summary address.
- As shown in the table below, convert the fourth segment of each of the three Hex prefixes (0010, 0020, 0030) into binary.

Hex	Binary			
:0010:	0000	0000	0001	0000
:0020:	0000	0000	0010	0000
:0030:	0000	0000	0011	0000

- The leading **10** bits for each of the three segments are identical and can also be used as part of the summary address.
- 48 bits + 10 bits gives us a /58 mask for the summary address.

Summary address = 2001:db8:a::/58

2. **Configure the aggregate-address command on SW1 to advertise a summary route.**

```
SW1(config)#router bgp 100
SW1(config-router)#address-family ipv6
SW1(config-router-af)#aggregate-address 2001:db8:a::/58
```

Verify by viewing the:

- BGP table on SW1 to see it has created the summary address.
- BGP table on SW2, SW3 and SW4 to verify the devices have received the summary address and still retain the longer prefixes.
- Specific prefix on SW1 to see it has assigned attributes aggregated and atomic-aggregate.
- Specific prefix on SW2, SW3 and SW4 to verify they know the router which summarized the prefixes.
- Specific prefix along with the longer-prefixes parameter to see the summary and all longer prefixes encompassed by the summary address.

```
SW1#show bgp ipv6
BGP table version is 65, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network          Next Hop              Metric LocPrf Weight Path
*> 2001:db8:a::/58 ::                                    32768 i
*> 2001:db8:a:10::/64
                    ::                             100   32768 i
*> 2001:db8:a:20::/64
                    ::                             100   32768 i
*> 2001:db8:a:30::/64
                    ::                             100   32768 i
Number of prefixes 4
```

```
SW2#show bgp ipv6
BGP table version is 63, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network          Next Hop         Metric LocPrf Weight Path
*> 2001:db8:a::/58
                  2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                        0             0 100 i
*> 2001:db8:a:10::/64
                  2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                        0             0 100 i
*> 2001:db8:a:20::/64
                  2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                        0             0 100 i
*> 2001:db8:a:30::/64
                  2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                        0             0 100 i
Number of prefixes 4
```

```
SW3#show bgp ipv6
BGP table version is 63, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network          Next Hop           Metric LocPrf Weight Path
*> 2001:db8:a::/58
                  2001:db8:20::1(fe80::eecd:6dff:fe20:c26b)
                                        0             0 200 100 i
*> 2001:db8:a:10::/64
                  2001:db8:20::1(fe80::eecd:6dff:fe20:c26b)
                                        0             0 200 100 i
*> 2001:db8:a:20::/64
                  2001:db8:20::1(fe80::eecd:6dff:fe20:c26b)
                                        0             0 200 100 i
*> 2001:db8:a:30::/64
                  2001:db8:20::1(fe80::eecd:6dff:fe20:c26b)
                                        0             0 200 100 i
Number of prefixes 4
```

```
SW4#show bgp ipv6
BGP table version is 63, local router ID is 4.4.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:a::/58
                 2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                          0             0 200 100 i
*> 2001:db8:a:10::/64
                 2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                          0             0 200 100 i
*> 2001:db8:a:20::/64
                 2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                          0             0 200 100 i
*> 2001:db8:a:30::/64
                 2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                          0             0 200 100 i
Number of prefixes 4
```

```
SW1#show bgp ipv6 2001:db8:a::/58
BGP routing table entry for 2001:db8:a::/58
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
  2001:db8:10::2
   Local, (aggregated by 100 1.1.1.1)
from :: (1.1.1.1)
   Origin IGP, localpref 100, weight 32768, valid, aggregated, local,
     atomic-aggregate, best
      Last update: Mon Aug 5 22:06:50 2013
```

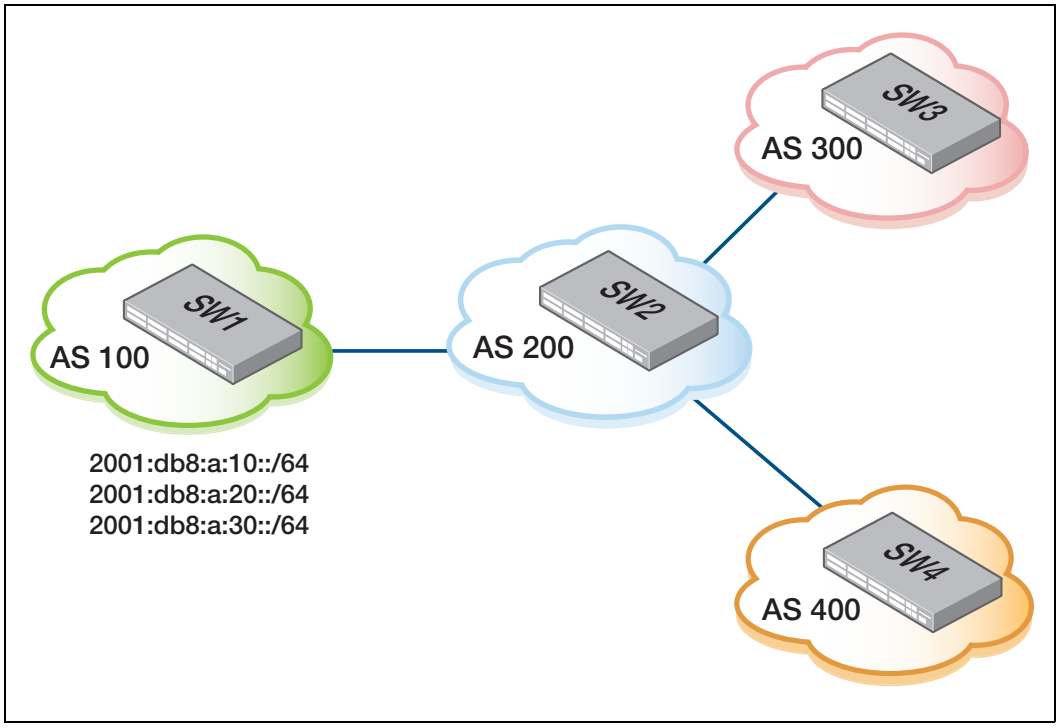
```
SW2#show bgp ipv6 2001:db8:a::/58
BGP routing table entry for 2001:db8:a::/58
Paths: (1 available, best #1, table Default-IP-Routing-Table)
   Advertised to non peer-group peers:
   2001:db8:20::2 2001:db8:30::2
   100, (aggregated by 100 1.1.1.1)
    2001:db8:10::1(fe80::eecd:6dff:fe52:b789) from 2001:db8:10::1 (1.1.1.1)
    (fe80::eecd:6dff:fe52:b789)
    Origin IGP metric 0, localpref 100, valid, external, atomic-aggregate, best
     Last update: Mon Aug 5 22:07:31 2013
```

```
SW3#show bgp ipv6 2001:db8:a::/58
BGP routing table entry for 2001:db8:a::/58
Paths: (1 available, best #1, table Default-IP-Routing-Table)
 Not advertised to any peer
 200 100, (aggregated by 100 1.1.1.1)
 2001:db8:20::1(fe80::eecd:6dff:fe20:c26b) from 2001:db8:20::1 (2.2.2.2)
 (fe80::eecd:6dff:fe20:c26b)
  Origin IGP metric 0, localpref 100, valid, external, atomic-aggregate, best
   Last update: Mon Aug 5 22:07:30 2013
```

```
SW4#show bgp ipv6 2001:db8:a::/58
BGP routing table entry for 2001:db8:a::/58
Paths: (1 available, best #1, table Default-IP-Routing-Table)
 Not advertised to any peer
  200 100, (aggregated by 100 1.1.1.1)
   2001:db8:30::1(fe80::eecd:6dff:fe20:c26b) from 2001:db8:30::1 (2.2.2.2)
   (fe80::eecd:6dff:fe20:c26b)
    Origin IGP metric 0, localpref 100, valid, external, atomic-aggregate, best
    Last update: Mon Aug 5 22:08:26 2013
```

```
SW2#show bgp ipv6 2001:db8:a::/58 longer-prefixes
BGP table version is 83, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network             Next Hop             Metric LocPrf Weight Path
*> 2001:db8:a::/58
                    2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                                 0             0 100 i
*> 2001:db8:a:10::/64
                    2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                                 0             0 100 i
*> 2001:db8:a:20::/64
                    2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                                 0             0 100 i
*> 2001:db8:a:30::/64
                    2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                                 0             0 100 i
Number of prefixes 4
```

Setup basic route aggregation in BGP4+ and suppress more specific prefixes from being advertised



Route aggregation is used to:

- Aggregate longer prefixes and advertise a single summary route.
- Decrease number of prefixes being advertised and reduce routing table size.

Objective

- Use the BGP4+ aggregate-address command under the IPv6 address family to advertise a summary route from SW1.
- More specific prefixes are suppressed in this configuration.

SW2, SW3, and SW4's BGP tables prior to route aggregation

```
SW2#show bgp ipv6
BGP table version is 23, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop                  Metric LocPrf Weight Path
*> 2001:db8:a:10::/64
                    2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                                   0             0 100 i
*> 2001:db8:a:20::/64
                    2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                                   0             0 100 i
*> 2001:db8:a:30::/64
                    2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                                   0             0 100 i
Number of prefixes 3
```

```
SW3#show bgp ipv6
BGP table version is 2, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop                Metric LocPrf Weight Path
*> 2001:db8:a:10::/64
                   2001:db8:20::1(fe80::eecd:6dff:fe20:c26b)
                                                   0            0 200 100 i
*> 2001:db8:a:20::/64
                   2001:db8:20::1(fe80::eecd:6dff:fe20:c26b)
                                                   0            0 200 100 i
*> 2001:db8:a:30::/64
                   2001:db8:20::1(fe80::eecd:6dff:fe20:c26b)
                                                   0            0 200 100 i
Number of prefixes 3
```

```
SW4#show bgp ipv6
BGP table version is 2, local router ID is 4.4.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network            Next Hop                  Metric LocPrf Weight Path
*> 2001:db8:a:10::/64
                    2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                                   0             0 200 100 i
*> 2001:db8:a:20::/64
                    2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                                   0             0 200 100 i
*> 2001:db8:a:30::/64
                    2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                                   0             0 200 100 i
Number of prefixes 3
```

1. Calculate the summary address with an appropriate subnet mask that matches the subnets to be summarized into a single prefix.

- List the prefixes to summarize:

  `2001:0db8:000a:0010::/64`

  `2001:0db8:000a:0020::/64`

  `2001:0db8:000a:0030::/64`

- Each digit in an IPv6 address is 4 bits. Each colon-separated segment of the IPv6 address is 16 bits.
- The first three identical segments of each prefix is `2001:0db8:000a`, that is **48** bits to be used as part of the summary address.
- As shown in the table below, convert the fourth segment of each of the three Hex prefixes (0010, 0020, 0030) into binary.

Hex	Binary			
:0010:	0000	0000	0001	0000
:0020:	0000	0000	0010	0000
:0030:	0000	0000	0011	0000

- The leading **10** bits for each of the three segments are identical and can also be used as part of the summary address.
- 48 bits + 10 bits gives us a /58 mask for the summary address.

Summary address = 2001:db8:a::/58

2. Configure the aggregate-address command on SW1 to advertise a summary route.

- The summary-only parameter suppressed longer prefixes from being advertised.

```
SW1(config)#router bgp 100
SW1(config-router)#address-family ipv6
SW1(config-router-af)#aggregate-address 2001:db8:a::/58 summary-only
```

Verify by viewing the:

- BGP table on SW1 to see it has created the summary address. (Notice the "s" next to the longer prefixes, showing they are suppressed.)
- BGP table on SW2, SW3 and SW4 to verify the devices have received the summary address only.
- Specific prefix on SW1 to see it has assigned attributes aggregated and atomic-aggregate.
- Specific prefix on SW2, SW3 and SW4 to verify they know the router which summarized the prefixes.

```
SW1#show bgp ipv6
BGP table version is 12, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network           Next Hop               Metric LocPrf Weight Path
*> 2001:db8:a::/58  ::                                    32768 i
s> 2001:db8:a:10::/64
                    ::                              100   32768 i
s> 2001:db8:a:20::/64
                    ::                              100   32768 i
s> 2001:db8:a:30::/64
                    ::                              100   32768 i
Number of prefixes 4
```

```
SW2#show bgp ipv6
BGP table version is 12, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network           Next Hop            Metric LocPrf Weight Path
*> 2001:db8:a::/58 2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                           0             0 100 i
Number of prefixes 1
```

```
SW3#show bgp ipv6
BGP table version is 12, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network           Next Hop             Metric LocPrf Weight Path
*> 2001:db8:a::/58 2001:db8:20::1(fe80::eecd:6dff:fe20:c26b)
                                            0             0 200 100 i
Number of prefixes 1
```

```
SW4#show bgp ipv6
BGP table version is 11, local router ID is 4.4.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network           Next Hop              Metric LocPrf Weight Path
*> 2001:db8:a::/58 2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                             0             0 200 100 i
Number of prefixes 1
```

```
SW1#show bgp ipv6 2001:db8:a::/58
BGP routing table entry for 2001:db8:a::/58
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
  2001:db8:10::2
  Local, (aggregated by 100 1.1.1.1)
from :: (1.1.1.1)
   Origin IGP, localpref 100, weight 32768, valid, aggregated, local, atomic-
aggregate, best
   Last update: Mon Aug 5 23:10:25 2013
```

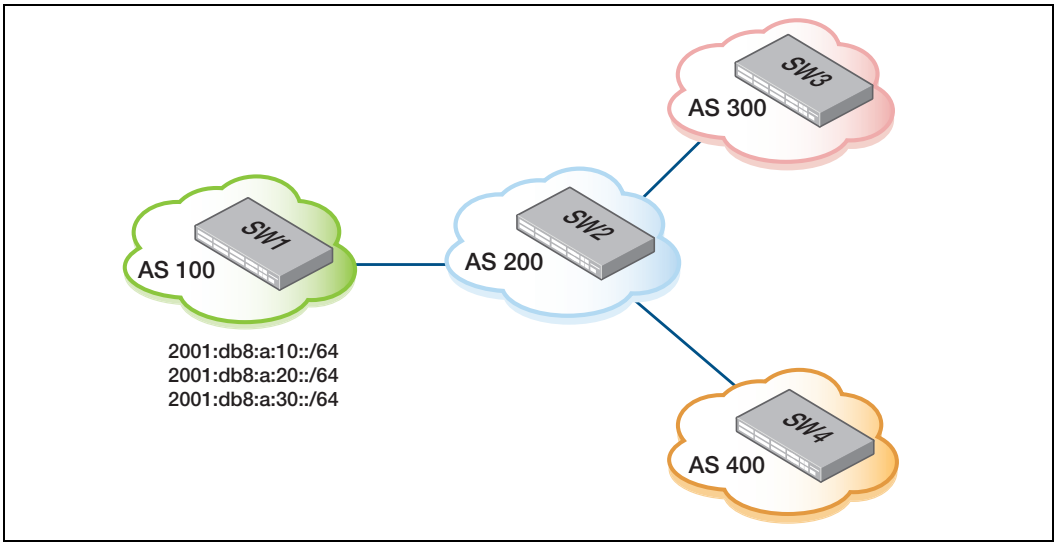
```
SW2#show bgp ipv6 2001:db8:a::/58
BGP routing table entry for 2001:db8:a::/58
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
  2001:db8:20::2 2001:db8:30::2
  100, (aggregated by 100 1.1.1.1)
    2001:db8:10::1(fe80::eecd:6dff:fe52:b789) from 2001:db8:10::1 (1.1.1.1)
     fe80::eecd:6dff:fe52:b789)
      Origin IGP metric 0, localpref 100, valid, external, atomic-aggregate, best
       Last update: Mon Aug 5 23:11:00 2013
```

```
SW3#show bgp ipv6 2001:db8:a::/58
BGP routing table entry for 2001:db8:a::/58
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  200 100, (aggregated by 100 1.1.1.1)
   2001:db8:20::1(fe80::eecd:6dff:fe20:c26b) from 2001:db8:20::1 (2.2.2.2)
   (fe80::eecd:6dff:fe20:c26b)
     Origin IGP metric 0, localpref 100, valid, external, atomic-aggregate, best
      Last update: Mon Aug 5 23:11:00 2013
```

```
SW4#show bgp ipv6 2001:db8:a::/58
BGP routing table entry for 2001:db8:a::/58
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  200 100, (aggregated by 100 1.1.1.1)
  2001:db8:30::1(fe80::eecd:6dff:fe20:c26b) from 2001:db8:30::1 (2.2.2.2)
  (fe80::eecd:6dff:fe20:c26b)
    Origin IGP metric 0, localpref 100, valid, external, atomic-aggregate, best
     Last update: Mon Aug 5 23:11:53 2013
```

Setup basic route aggregation to aggregate routes received from a neighboring AS

Aggregating routes learned from a BGP4+ peer. SW2 will aggregate longer prefixes learned from SW1.



Objective

- Use the BGP4+ aggregate-address command under the IPv6 address family on SW2 to aggregate the routes: 2001:db8:a:10::/64, 2001:db8:a:20::/64 and 2001:db8:a:30::/64, learned from SW1 into a single summary route.
- Suppress longer prefixes and advertise the summary route to other BGP peers connected to SW2.
- Analyse the potential issue of black hole routing.
- Advertise an AS set for the summary route to include the ASN where the longer prefixes originated.

Quick guide on BGP aggregation attributes

AGGREGATOR: identifies the router which summarized the routes.

ATOMIC-AGGREGATE: Indicates the NLRI should not be de-aggregated.

AS set

- An AS set is an unordered list of ASNs from all of the routes being aggregated.
- This is used to prevent AS loops occurring as the BGP speaker who advertised the longer prefixes will deny the summary route due to seeing its own ASN in the AS_PATH.

Summary route advertisement conditions

- There must be at least one prefix residing within the summary address, for the summary route to be advertised to peers.

SW2, SW3, and SW4's BGP tables prior to route aggregation

```
SW2#show bgp ipv6
BGP table version is 23, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network              Next Hop              Metric LocPrf Weight Path
*> 2001:db8:a:10::/64
                     2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                                0             0 100 i
*> 2001:db8:a:20::/64
                     2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                                0             0 100 i
*> 2001:db8:a:30::/64
                     2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                                0             0 100 i
Number of prefixes 3
```

```
SW3#show bgp ipv6
BGP table version is 2, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network              Next Hop              Metric LocPrf Weight Path
*> 2001:db8:a:10::/64
                     2001:db8:20::1(fe80::eecd:6dff:fe20:c26b)
                                                0             0 200 100 i
*> 2001:db8:a:20::/64
                     2001:db8:20::1(fe80::eecd:6dff:fe20:c26b)
                                                0             0 200 100 i
*> 2001:db8:a:30::/64
                     2001:db8:20::1(fe80::eecd:6dff:fe20:c26b)
                                                0             0 200 100 i
Number of prefixes 3
```

```
SW4#show bgp ipv6
BGP table version is 2, local router ID is 4.4.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network              Next Hop              Metric LocPrf Weight Path
*> 2001:db8:a:10::/64
                     2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                                0             0 200 100 i
*> 2001:db8:a:20::/64
                     2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                                0             0 200 100 i
*> 2001:db8:a:30::/64
                     2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                                0             0 200 100 i
Number of prefixes 3
```

1. Calculate the summary address with an appropriate subnet mask. This single summarized route matches all three prefixes.

- List the prefixes to summarize:

  `2001:0db8:000a:0010::/64`

  `2001:0db8:000a:0020::/64`

  `2001:0db8:000a:0030::/64`

- Each digit in an IPv6 address is 4 bits. Each colon-separated segment of the IPv6 address is 16 bits.
- The first three identical segments of each prefix is `2001:0db8:000a`, that is **48** bits to be used as part of the summary address.
- As shown in the table below, convert the fourth segment of each of the three Hex prefixes (0010, 0020, 0030) into binary.

Hex	Binary			
:0010:	0000	0000	0001	0000
:0020:	0000	0000	0010	0000
:0030:	0000	0000	0011	0000

- The leading **10** bits for each of the three segments above are identical and can also be used as part of the summary address.
- 48 bits +10 bits gives us a /58 mask for the summary address.

Summary address = 2001:db8:a::/58

2. Configure the aggregate-address command on SW2 to advertise a summary route.

- SW2 will advertise a summary route to peers and suppress the more specific prefixes learned from SW1.
- The summary-only parameter suppresses longer prefixes from being advertised.

```
SW2(config)#router bgp 200
SW2(config-router)#address-family ipv6
SW2(config-router-af)#aggregate-address 2001:db8:a::/58 summary-only
```

3. Verify and explore the potential issue of black hole routing.

We can see SW2 has aggregated the routes and is advertising a summary route. However, a new issue has been created where SW2 will also advertise the summary route back to SW1.

SW1 will accept and install this prefix as only SW2's ASN is in the AS_PATH. This result occurs because SW1 does not discard the prefix because its own ASN is not in the AS_PATH.

```
SW2#show bgp ipv6
BGP table version is 16, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network            Next Hop                Metric LocPrf Weight Path
*> 2001:db8:a::/58 ::                                      32768 i
s> 2001:db8:a:10::/64
                   2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                                    0             0 100 i
s> 2001:db8:a:20::/64
                   001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                                    0             0 100 i
s> 2001:db8:a:30::/64
                   2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                                    0             0 100 i
Number of prefixes 4
```

Note: SW2 advertises the route as originating from its own AS.

```
SW2#show bgp ipv6 2001:db8:a::/58
BGP routing table entry for 2001:db8:a::/58
 Paths: (1 available, best #1, table Default-IP-Routing-Table)
 Advertised to non peer-group peers:
 2001:db8:10::1 2001:db8:20::2 2001:db8:30::2
 Local, (aggregated by 200 2.2.2.2)
from :: (2.2.2.2)
 Origin IGP, localpref 100, weight 32768, valid, aggregated, local, atomic-
aggregate, best
 Last update: Mon Aug 5 23:47:19 2013
```

- SW1 has learned the summary route from SW2.
- AS_PATH shows summary has originated from AS 200.
- SW1 installs the prefix as it believes there is no loop.

```
SW1#show bgp ipv6
BGP table version is 15, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network               Next Hop            Metric LocPrf Weight Path
*> 2001:db8:a::/58    2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                                  0             0 200 i
*> 2001:db8:a:10::/64
                      ::
                                                                100 32768 i
*> 2001:db8:a:20::/64
                      ::
                                                                100 32768 i
*> 2001:db8:a:30::/64
                      ::
                                                                100 32768 i
Number of prefixes 4
```

How a black hole route could form

If one of the specific prefixes advertised by SW1 are no longer reachable by SW1:

- The next best path to that prefix is the less specific summary route learned from SW2.
- SW1 would route traffic for the destination prefix to SW2.
- SW2 will no longer have the longer prefix originally advertised by SW1.
- SW2 will drop traffic destined for the longer prefix.
- A black hole route has formed for all traffic destined to the specific prefix.

4. Mitigate the potential issue of a black hole route by advertising the AS set.

- Use the as-set parameter on SW2 to advertise an AS set to peers receiving the summary route.

```
SW2(config)#router bgp 200
SW2(config-router)#address-family ipv6
SW2(config-router-af)#aggregate-address 2001:db8:a::/58 as-set summary-
    only
```

Verify by viewing the BGP table on:

- SW1 to view the prefix has been denied.
- SW2 to view the suppressed longer prefixes and that the originating AS is 100.
- SW3 and SW4 to see the originating ASN for the route is included (unordered-shown with the { }) in the AS_PATH.

```
SW1#show bgp ipv6
BGP table version is 62, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network           Next Hop               Metric LocPrf Weight Path
*> 2001:db8:a:10::/64
                    ::                                100  32768 i
*> 2001:db8:a:20::/64
                    ::                                100  32768 i
*> 2001:db8:a:30::/64
                    ::                                100  32768 i
Number of prefixes 3
```

```
SW2#show bgp ipv6
BGP table version is 66, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network              Next Hop           Metric LocPrf Weight Path
*> 2001:db8:a::/58 ::                                    32768 {100} {100} i
s> 2001:db8:a:10::/64
                     2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                              0             0 100 i
s> 2001:db8:a:20::/64
                     2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                              0             0 100 i
s> 2001:db8:a:30::/64
                     2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                              0             0 100 i
Number of prefixes 4
```

```
SW2#show bgp ipv6 2001:db8:a::/58
BGP routing table entry for 2001:db8:a::/58
Paths: (1 available, best #1, table Default-IP-Routing-Table)
   Advertised to non peer-group peers:
   2001:db8:10::1 2001:db8:20::2 2001:db8:30::2
   {100} {100}, (aggregated by 200 2.2.2.2)
from :: (2.2.2.2)
   Origin IGP, localpref 100, weight 32768, valid, aggregated, local, atomic-
aggregate, best
    Last update: Tue Aug 6 00:46:21 2013
```

```
SW3#show bgp ipv6
BGP table version is 3, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network            Next Hop           Metric LocPrf Weight Path
*> 2001:db8:a::/58 2001:db8:20::1(fe80::eecd:6dff:fe20:c26b)
                                          0             0 200 {100} {100} i
Number of prefixes 1
```

```
SW3#show bgp ipv6 2001:db8:a::/58
BGP routing table entry for 2001:db8:a::/58
Paths: (1 available, best #1, table Default-IP-Routing-Table)
   Not advertised to any peer
   200 {100} {100}, (aggregated by 200 2.2.2.2)
    2001:db8:20::1(fe80::eecd:6dff:fe20:c26b) from 2001:db8:20::1 (2.2.2.2)
    (fe80::eecd:6dff:fe20:c26b)
     Origin IGP metric 0, localpref 100, valid, external, atomic-aggregate, best
     Last update: Tue Aug 6 00:46:17 2013
```

```
SW4#show bgp ipv6
BGP table version is 3, local router ID is 4.4.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:a::/58 2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                                0             0 200 {100} {100} i
Number of prefixes 1
```

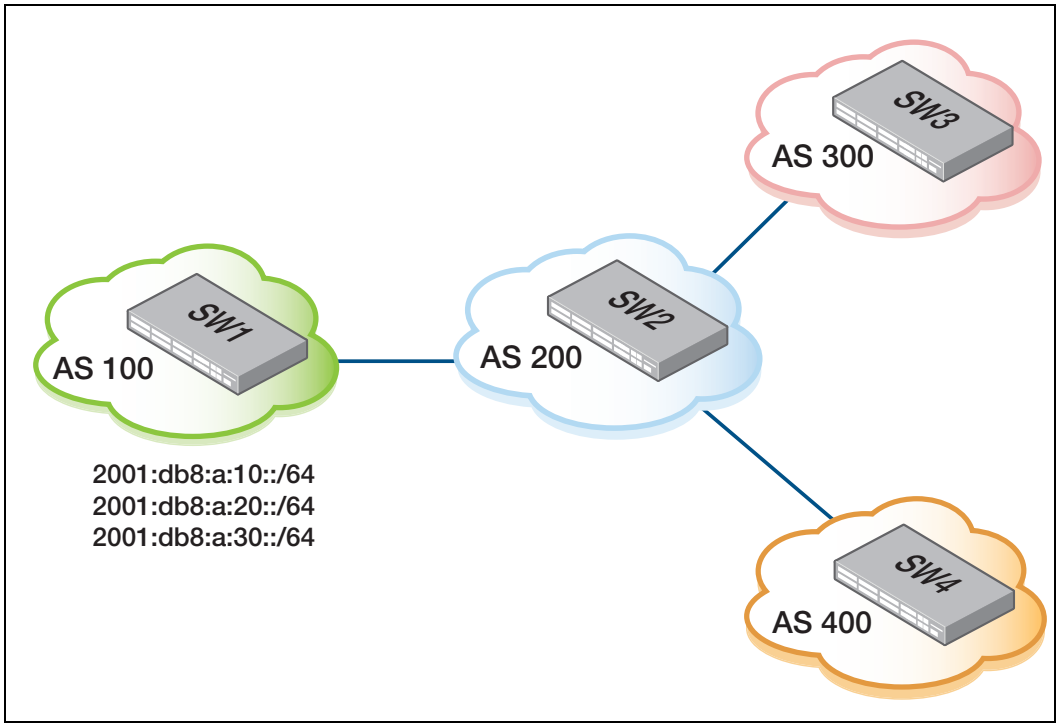
```
SW4#show bgp ipv6 2001:db8:a::/58
BGP routing table entry for 2001:db8:a::/58
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  200 {100} {100}, (aggregated by 200 2.2.2.2)
    2001:db8:30::1(fe80::eecd:6dff:fe20:c26b) from 2001:db8:30::1 (2.2.2.2)
    (fe80::eecd:6dff:fe20:c26b)
     Origin IGP metric 0, localpref 100, valid, external, atomic-aggregate, best
      Last update: Tue Aug 6 00:47:05 2013
```

Verifying with debug

- On SW1 use debug to view BGP has denied the summary route as it contains AS 100 in the AS_PATH.

```
00:46:34 SW1 BGP[1500]: 2001:db8:10::2-Outgoing [DECODE] Msg-Hdr: type 2, length
112
00:46:34 SW1 BGP[1500]: 2001:db8:10::2-Outgoing [DECODE] Msg-Hdr: Requesting
immediate Read (93)
00:46:34 SW1 BGP[1500]: 2001:db8:10::2-Outgoing [DECODE] Update: Starting UPDATE
decoding..Bytes To Read (93), msg_size (93)
00:46:34 SW1 BGP[1500]: 2001:db8:10::2-Outgoing [FSM] State: Established Event:
27
00:46:34 SW1 BGP[1500]: 2001:db8:10::2-Outgoing [RIB] Update: Prefix
2001:db8:a::/58 denied due to as-path contains our own AS
```

Setup an unsuppress map to advertise longer prefixes suppressed by an aggregate address



Route aggregation is used to:

- Aggregate longer prefixes and advertise a single summary route.
- Decrease number of prefixes being advertised and reduce routing table size.

An unsuppress map is used to:

- Unsuppress suppressed routes (which have longer prefix masks) to specific peers.
- Advertise more specific prefixes (with the longer masks) to peers.

Note: The unsuppress map does NOT prevent the summary route being advertised. If this is desired, route filtering must also be applied.

Objectives

- Suppress the longer prefixes being advertised to SW3 from SW2.
- Unsuppress the longer prefixes being advertised to SW4 from SW2.

SW2, SW3, and SW4's BGP tables prior to route aggregation

```
SW2#show bgp ipv6
BGP table version is 23, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network             Next Hop                    Metric LocPrf Weight Path
*> 2001:db8:a:10::/64
                    2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                                     0             0 100 i
*> 2001:db8:a:20::/64
                    2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                                     0             0 100 i
*> 2001:db8:a:30::/64
                    2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                                     0             0 100 i
Number of prefixes 3
```

```
SW3#show bgp ipv6
BGP table version is 2, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network            Next Hop                     Metric LocPrf Weight Path
*> 2001:db8:a:10::/64
                    2001:db8:20::1(fe80::eecd:6dff:fe20:c26b)
                                                     0             0 200 100 i
*> 2001:db8:a:20::/64
                    2001:db8:20::1(fe80::eecd:6dff:fe20:c26b)
                                                     0             0 200 100 i
*> 2001:db8:a:30::/64
                    2001:db8:20::1(fe80::eecd:6dff:fe20:c26b)
                                                     0             0 200 100 i
Number of prefixes 3
```

```
SW4#show bgp ipv6
BGP table version is 2, local router ID is 4.4.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network            Next Hop                   Metric LocPrf Weight Path
*> 2001:db8:a:10::/64
                    2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                                    0             0 200 100 i
*> 2001:db8:a:20::/64
                    2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                                    0             0 200 100 i
*> 2001:db8:a:30::/64
                    2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                                    0             0 200 100 i
Number of prefixes 3
```

1. Calculate the summary address with an appropriate subnet mask that matches the subnets to be summarized into a single prefix.

- List the prefixes to summarize:

  `2001:0db8:000a:0010::/64`

  `2001:0db8:000a:0020::/64`

  `2001:0db8:000a:0030::/64`

- Each digit in an IPv6 address is 4 bits. Each colon-separated segment of the IPv6 address is 16 bits.
- The first three identical segments of each prefix is `2001:0db8:000a`, that is **48** bits to be used as part of the summary address.
- As shown in the table below, convert the fourth segment of each of the three Hex prefixes (0010, 0020, 0030) into binary.

Hex	Binary			
:0010:	0000	0000	0001	0000
:0020:	0000	0000	0010	0000
:0030:	0000	0000	0011	0000

- The leading **10** bits for each of the three segments are identical and can also be used as part of the summary address.
- 48 bits +10 bits gives us a /58 mask for the summary address.

Summary address = 2001:db8:a::/58

2. Configure the aggregate-address command on SW2 to advertise a summary route.

```
SW2(config)#router bgp 100
SW2(config-router)#address-family ipv6
SW2(config-router-af)#aggregate-address 2001:db8:a::/58 as-set summary-
   only
```

3. Configure an unsuppress map.

- Create an IPv6 ACL or prefix list to classify the suppressed prefixes.

```
SW2(config)#ipv6 prefix-list BGP_UNSUPPRESS_LIST seq 10 permit
   2001:db8:a:10::/64
SW2(config)#ipv6 prefix-list BGP_UNSUPPRESS_LIST seq 20 permit
   2001:db8:a:20::/64
SW2(config)#ipv6 prefix-list BGP_UNSUPPRESS_LIST seq 30 permit
   2001:db8:a:30::/64
```

- Create a route map to match on the ACL or prefix list.

```
SW2(config)#route-map BGP_UNSUPPRESS_MAP permit 10
SW2(config-route-map)#match ipv6 address prefix-list
   BGP_UNSUPPRESS_LIST
```

4. Apply the unsuppress map to the peering session to SW4.

```
SW2(config)#router bgp 200
SW2(config-router)#address-family ipv6
SW2(config-router-af)#neighbor 2001:db8:30::2 unsuppress-map
   BGP_UNSUPPRESS_MAP
```

5. Clear the BGP session to SW4 to apply the changes.

```
SW2#clear bgp ipv6 2001:db8:30::2
```

Verify by viewing the:

- BGP table on SW3 and SW4. (SW3 should only have learned the summary route, and SW4 should learn the summary and longer prefixes).
- BGP neighbor information for SW4 on SW2. The unsuppress map route map policy should be present under the IPv6 address family information.
- Matching prefixes for the unsuppress map on SW2.

```
SW3#show bgp ipv6
BGP table version is 3, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:a::/58 2001:db8:20::1(fe80::eecd:6dff:fe20:c26b)
                                                0             0 200 {100} {100} i
Number of prefixes 1
```

```
SW4#show bgp ipv6
BGP table version is 6, local router ID is 4.4.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network            Next Hop            Metric LocPrf Weight Path
*> 2001:db8:a::/58 2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                                0             0 200 {100} {100} i
*> 2001:db8:a:10::/64
                   2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                                0             0 200 100 i
*> 2001:db8:a:20::/64
                   2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                                0             0 200 100 i
*> 2001:db8:a:30::/64
                   2001:db8:30::1(fe80::eecd:6dff:fe20:c26b)
                                                0             0 200 100 i
Number of prefixes 4
```

```
SW2#show bgp ipv6 neighbors 2001:db8:30::2
BGP neighbor is 2001:db8:30::2, remote AS 400, local AS 200, external link
  BGP version 4, remote router ID 4.4.4.4
  BGP state = Established, up for 00:00:55
  Last read 00:00:55, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    4-Octet ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 29 messages, 0 notifications, 0 in queue
  Sent 35 messages, 4 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
For address family: IPv6 Unicast
 BGP table version 3, neighbor version 3
 Index 3, Offset 0, Mask 0x8
 Community attribute sent to this neighbor (both)
 Outbound path policy configured
 Route map for selective unsuppress is *BGP_UNSUPPRESS_MAP
 0 accepted prefixes
 4 announced prefixes

 Connections established 4; dropped 3
Local host: 2001:db8:30::1, Local port: 53823
Foreign host: 2001:db8:30::2, Foreign port: 179
Nexthop: 2.2.2.2
Nexthop global: 2001:db8:30::1
Nexthop local: fe80::eecd:6dff:fe20:c26b
BGP connection: shared network
Last Reset: 00:00:55, due to BGP Notification sent
Notification Error Message: (Cease/Other Configuration Change.)
```

```
SW2#show bgp ipv6 route-map BGP_UNSUPPRESS_MAP
BGP table version is 5, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
s> 2001:db8:a:10::/64
                    2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                           0             0 100 i
s> 2001:db8:a:20::/64
                    2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                           0             0 100 i
s> 2001:db8:a:30::/64
                    2001:db8:10::1(fe80::eecd:6dff:fe52:b789)
                                           0             0 100 i
Number of prefixes 3
```

This chapter covers the following topics:

- Using local preference to change the preferred path for selected prefix(es) - Inbound method
- Using local preference to change the preferred path for selected prefix(es) - Outbound method
- Multi-Exit Discriminator (MED)
- Using deterministic-med to ensure MED comparison from the same AS
- Comparing MED from different AS to change the path preference
- Using deterministic-med and always-compare-med together to change the path preference
- Using the BGP backdoor feature to change path preference without altering the default BGP AD
- Using weight to change path preference for all prefix(es) received into a router

CHAPTER 6

Path Control

BGP4+ uses an algorithm to compare routes on the basis of local preference. The local preference indicates the preferred path when there are multiple paths to the same destination. The path with the higher preference is preferred.

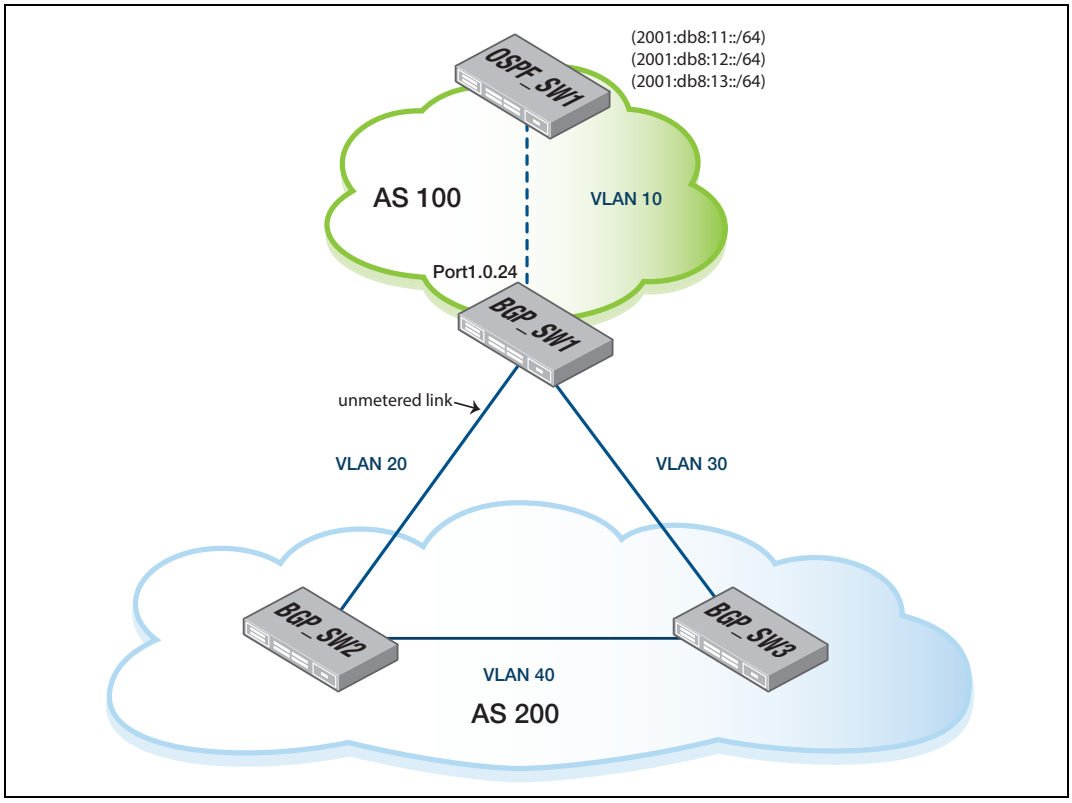
You can define a particular path as more preferable or less preferable than other paths by changing the default local preference value of 100.

The following scenarios present alternative methods to achieve selective path control.

Using local preference to change the preferred path for selected prefix(es) - Inbound method

Scenario

- BGP_SW1 resides in AS 100.
- BGP_SW2 and BGP_SW3 reside in AS 200.
- The link between BGP_SW1 and BGP_SW2 is unmetered and incurs no additional cost.
- The link between BGP_SW1 and BGP_SW3 is metered and incurs additional usage costs.



Objective

The objective is to ensure the path selected by BGP_SW3 to networks 2001:db8:11::/64, 2001:db8:12::/64 and 2001:db8:13::/64 (11,12,13) originating from AS 100 (OSPF_SW1), is via the unmetered link between BGP_SW1 and BGP_SW2, (incurs no additional cost), instead of via the more direct link via VLAN 30.

BGP_SW3 learns about the networks 11,12,13 from both BGP peers BGP_SW1 and BGP_SW2. By default, BGP_SW3 chooses the path via eBGP neighbor BGP_SW1. Since the Local Preference values are the same, the tie breaker for routing path selection used is BGP's preference for the eBGP path over the iBGP path - based on the BGP best path selection algorithm.

To achieve this non-default objective, BGP_SW3 is to apply a local preference value to prefixes received in BGP updates from iBGP neighbor BGP_SW2. Applying a higher local preference on BGP_SW3 for the received prefixes, BGP will then determine the preferred path is via iBGP neighbor BGP_SW2. This will supersede the current preference for eBGP routing over an iBGP routing.

To achieve this non-default prerequisite

The BGP next-hop-self option must be configured for iBGP neighbors BGP_SW2 and BGP_SW3. This is because the iBGP neighbors will advertise the externally learned prefixes from AS 100, with the Next-Hop address of the external neighbor BGP_SW1. In this case, we need to use BGP_SW2's address as the Next Hop for prefixes advertised from BGP_SW2 to BGP_SW3 and vice-versa for prefixes advertised from BGP_SW3 to BGP_SW2.

This should also be configured on BGP_SW3 as well, in case the path to AS 100 via BGP_SW2 goes down.

```
BGP_SW2(config)#router bgp 200
BGP_SW2(config-router)#address-family ipv6
BGP_SW2(config-router-af)#neighbor 2001:db8:3::1 next-hop-self
BGP_SW2(config-router-af)#end
BGP_SW2#clear bgp ipv6 *

BGP_SW3(config)#router bgp 200
BGP_SW3(config-router)#address-family ipv6
BGP_SW3(config-router-af)#neighbor 2001:db8:2::1 next-hop-self
BGP_SW3(config-router-af)#end
BGP_SW3#clear bgp ipv6 *
```

View of BGP_SW3's BGP table PRIOR to Local Preference adjustment

BGP_SW3 currently uses the eBGP path via the directly connected (metered) link to AS 100, as the Local Preference values are the same.

Because the Local Preference values are the same, the tie breaker is BGP's preference for the eBGP path over the iBGP path.

We want to ensure BGP_SW3 selects the internal path via iBGP neighbor 2001:db8:2::1 (via BGP_SW2) instead of via the eBGP neighbor 2001:db8:1::1 (BGP_SW1).

Here is the BGP_SW3 table prior to Local Preference adjustment:

```
BGP_SW3#show bgp ipv6
BGP table version is 3, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
* i2001:db8:11::/64 2001:db8:2::1           20    100      0 100 ?
*>                  2001:db8:1::1           20             0 100 ?
* i2001:db8:12::/64 2001:db8:2::1           20    100      0 100 ?
*>                  2001:db8:1::1           20             0 100 ?
* i2001:db8:13::/64 2001:db8:2::1           20    100      0 100 ?
*>                  2001:db8:1::1           20             0 100 ?
```

1. Create an Access List or a Prefix List to classify the three networks 11,12, and 13.

Option1: (A more efficient prefix network and length can be defined if desired.)

```
BGP_SW3(config)#ipv6 prefix-list REMOTE_NETWORK_LIST seq 10 permit
   2001:db8:11::/64
BGP_SW3(config)#ipv6 prefix-list REMOTE_NETWORK_LIST seq 20 permit
   2001:db8:12::/64
BGP_SW3(config)#ipv6 prefix-list REMOTE_NETWORK_LIST seq 30 permit
   2001:db8:13::/64
```

Option 2:

```
BGP_SW3(config)#ipv6 access-list standard REMOTE_NETWORK_LIST permit
   2001:db8:11::/64
BGP_SW3(config)#ipv6 access-list standard REMOTE_NETWORK_LIST permit
   2001:db8:12::/64
BGP_SW3(config)#ipv6 access-list standard REMOTE_NETWORK_LIST permit
   2001:db8:13::/64
```

2. Create a route map to match on the ACL or Prefix List and alter the local preference value for the matching prefix(es).

- Sequence 10 will apply local preference value of 150 to prefixes matching the ACL / Prefix List.
- Sequence 20 will permit all other prefixes, leaving the default value unchanged.

Option 1: (ACL used)

```
BGP_SW3(config)#route-map REMOTE_NETWORK permit 10
BGP_SW3(config-route-map)#match ipv6 address REMOTE_NETWORK_LIST
BGP_SW3(config-route-map)#set local-preference 150
BGP_SW3(config-route-map)#route-map REMOTE_NETWORK permit 20
```

Option 2: (Prefix List used)

```
BGP_SW3(config)#route-map REMOTE_NETWORK permit 10
BGP_SW3(config-route-map)#match ipv6 address prefix-list
   REMOTE_NETWORK_LIST
BGP_SW3(config-route-map)#set local-preference 150
BGP_SW3(config-route-map)#route-map REMOTE_NETWORK permit 20
```

3. Apply the route map for prefixes received in BGP updates on BGP_SW3 in AS 200.

The route map will apply for prefixes received by iBGP neighbor 2001:db8:2::1.

```
BGP_SW3(config)#router bgp 200
BGP_SW3(config-router)#address-family ipv6
BGP_SW3(config-router)#neighbor 2001:db8:2::1 route-map REMOTE_NETWORK
   in
```

4. Clear the BGP session to neighbor 2001:db8:2::1 (BGP_SW3 in AS 200) to apply the change.

```
BGP_SW2#clear bgp ipv6 2001:db8:3::1
```

Verify by:

- Using the show bgp ipv6 command and specifying a network to verify specific prefixes.
- Using the BGP table seen on BGP_SW3. Prefixes advertised from BGP_SW2 have a local preference value of 150.
- Using the routing table seen on BGP_SW3.
- Viewing the BGP neighbor output (Route map for incoming advertisements).
- Viewing the route map.
- Viewing the prefix list detail (Hit count per sequence).

```
BGP_SW3#show bgp ipv6 2001:db8:11::/64
BGP routing table entry for 2001:db8:11::/64
Paths: (2 available, best #1, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
  2001:db8:1::1
  AS-path: 100
    2001:db8:2::1 from 2001:db8:2::1 (2.2.2.2)
      Origin incomplete metric 20, localpref 150, valid, internal, best
      Last update: Wed May 15 10:11:52 2013
  AS-path: 100
    2001:db8:1::1 from 2001:db8:1::1 (1.1.1.1)
      Origin incomplete metric 20, localpref 100, valid, external
       Last update: Wed May 15 10:04:53 2013
```

```
BGP_SW3#show bgp ipv6
BGP table version is 12, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*>i2001:db8:11::/64 2001:db8:2::1           20    150      0 100 ?
*                   2001:db8:1::1           20             0 100 ?
*>i2001:db8:12::/64 2001:db8:2::1           20    150      0 100 ?
*                   2001:db8:1::1           20             0 100 ?
*>i2001:db8:13::/64 2001:db8:2::1           20    150      0 100 ?
*                   2001:db8:1::1           20             0 100 ?
Number of prefixes 3
```

```
BGP_SW3#show route-map
route-map REMOTE_NETWORK, permit, sequence 10
  Match clauses:
    ipv6 address prefix-list REMOTE_NETWORK_LIST
  Set clauses:
    local-preference 150
route-map REMOTE_NETWORK, permit, sequence 20
  Match clauses:
  Set clauses:
```

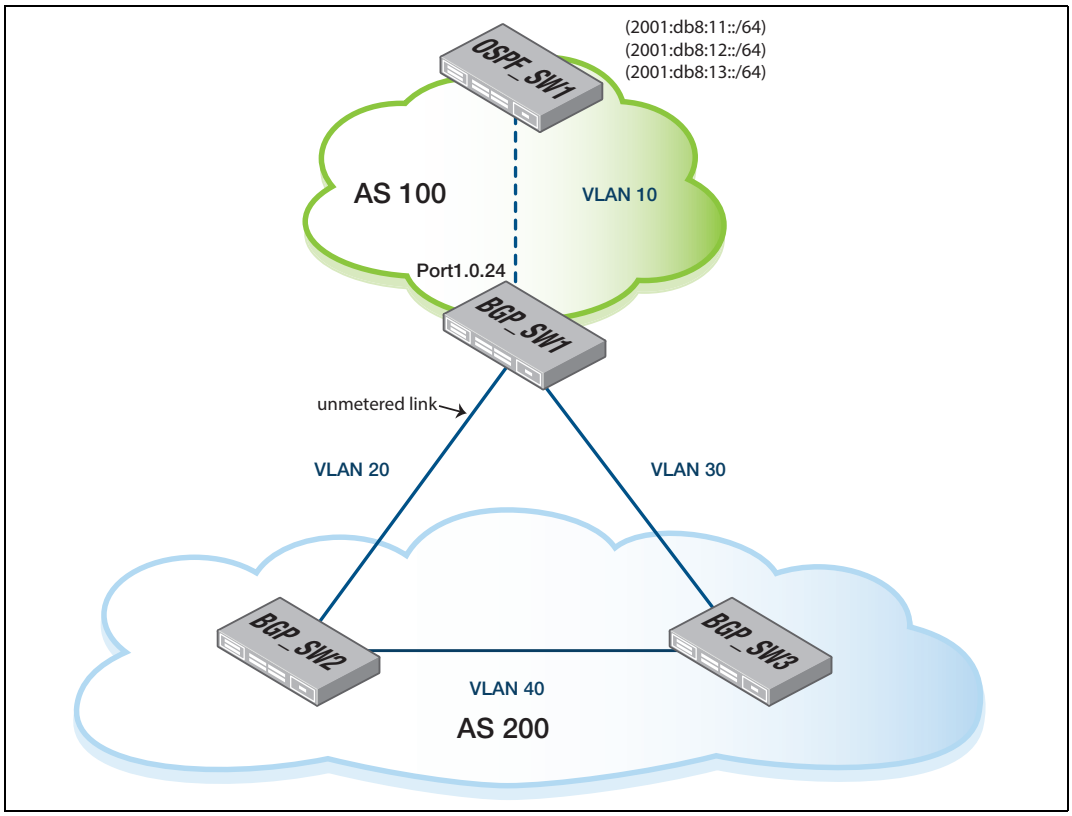
```
BGP_SW3#show ipv6 prefix-list detail
Prefix-list with the last deletion/insertion: REMOTE_NETWORK_LIST
ipv6 prefix-list REMOTE_NETWORK_LIST:
   count: 3, range entries: 0, sequences: 10 - 30
    ripd:
     seq 10 permit 2001:db8:11::/64 (hit count: 0, refcount: 0)
     seq 20 permit 2001:db8:12::/64 (hit count: 0, refcount: 0)
     seq 30 permit 2001:db8:13::/64 (hit count: 0, refcount: 0)
    ripngd:
     seq 10 permit 2001:db8:11::/64 (hit count: 0, refcount: 0)
     seq 20 permit 2001:db8:12::/64 (hit count: 0, refcount: 0)
     seq 30 permit 2001:db8:13::/64 (hit count: 0, refcount: 0)
    ospfd:
     seq 10 permit 2001:db8:11::/64 (hit count: 0, refcount: 0)
     seq 20 permit 2001:db8:12::/64 (hit count: 0, refcount: 0)
     seq 30 permit 2001:db8:13::/64 (hit count: 0, refcount: 0)
    ospf6d:
     seq 10 permit 2001:db8:11::/64 (hit count: 0, refcount: 0)
     seq 20 permit 2001:db8:12::/64 (hit count: 0, refcount: 0)
     seq 30 permit 2001:db8:13::/64 (hit count: 0, refcount: 0)
    bgpd:
     seq 10 permit 2001:db8:11::/64 (hit count: 1, refcount: 3)
     seq 20 permit 2001:db8:12::/64 (hit count: 1, refcount: 2)
     seq 30 permit 2001:db8:13::/64 (hit count: 1, refcount: 1)
```

BGP_SW3 configuration

```
BGP_SW3#

!
hostname BGP_SW3
!
vlan database
 vlan 30,40 state enable
!
interface port1.0.1
 switchport access vlan 30
!
interface port1.0.2
 switchport access vlan 40
!
interface lo
 ipv6 address 2001:db8:3::1/64
!
interface vlan30
 ipv6 address 2001:db8:30::2/64
!
interface vlan40
 ipv6 address 2001:db8:40::2/64
!
router bgp 200
 bgp router-id 3.3.3.3
 no bgp default ipv4-unicast
 neighbor 2001:db8:1::1 remote-as 100
 neighbor 2001:db8:1::1 ebgp-multihop 2
 neighbor 2001:db8:1::1 update-source lo
 neighbor 2001:db8:2::1 remote-as 200
 neighbor 2001:db8:2::1 update-source lo
 !
 address-family ipv6
 neighbor 2001:db8:1::1 activate
 neighbor 2001:db8:2::1 activate
 neighbor 2001:db8:2::1 next-hop-self
 neighbor 2001:db8:2::1 route-map REMOTE_NETWORK in
 exit-address-family
!
ipv6 route 2001:db8:1::1/128 fe80::200:cdff:fe2a:eaf1 vlan30
ipv6 route 2001:db8:2::1/128 fe80::209:41ff:fefb:c323 vlan40
!
ipv6 prefix-list REMOTE_NETWORK_LIST seq 10 permit 2001:db8:11::/64
ipv6 prefix-list REMOTE_NETWORK_LIST seq 20 permit 2001:db8:12::/64
ipv6 prefix-list REMOTE_NETWORK_LIST seq 30 permit 2001:db8:13::/64
!
route-map REMOTE_NETWORK permit 10
 match ipv6 address prefix-list REMOTE_NETWORK_LIST
 set local-preference 150
!
route-map REMOTE_NETWORK permit 20
!
end
```

Using local preference to change the preferred path for selected prefix(es) - Outbound method

Scenario



- BGP_SW1 resides in AS 100.
- BGP_SW2 and BGP_SW3 reside in AS 200.
- The link between BGP_SW1 and BGP_SW2 is unmetered and incurs no additional cost.
- The link between BGP_SW1 and BGP_SW3 is metered and incurs additional usage costs.

Objective

The objective is to ensure the path selected by BGP_SW3 to networks 11,12,13 located in AS 100 is via the link between BGP_SW1 and BGP_SW2. To achieve this objective, BGP_SW2 is to advertise a Local Preference of 150 to iBGP neighbor BGP_SW3 for AS 100 prefixes in BGP Updates.

By advertising a higher Local Preference for the selected prefixes, BGP will then determine that the internal path via BGP_SW2 is the preferred path. This will supersede the current preference for eBGP route over an iBGP route.

Prerequisite

The BGP next-hop-self option must be configured for iBGP neighbors BGP_SW2 and BGP_SW3. This is because the iBGP neighbors will advertise the prefixes learned from BGP_SW1 in AS 100 to each other, with the Next Hop of BGP_SW1. In this case we need to use BGP_SW2's address as the Next Hop for prefixes advertised to BGP_SW3.

This should also be configured on BGP_SW3 as well, incase the link via BGP_SW2 goes down.

```
BGP_SW2(config)#router bgp 200
BGP_SW2(config-router)#address-family ipv6
BGP_SW2(config-router-af)#neighbor 2001:db8:3::1 next-hop-self
BGP_SW2(config-router-af)#end
BGP_SW2#clear bgp ipv6 *

BGP_SW3(config)#router bgp 200
BGP_SW3(config-router)#address-family ipv6
BGP_SW3(config-router-af)#neighbor 2001:db8:2::1 next-hop-self
BGP_SW3(config-router-af)#end
BGP_SW3#clear bgp ipv6 *
```

View of BGP_SW2's BGP table prior to Local Preference adjustment

Currently BGP_SW2 takes the desired path via the unmetered link.

```
BGP_SW2#show bgp ipv6
BGP table version is 5, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
* i2001:db8:11::/64 2001:db8:3::1           20    100      0 100 ?
*>                  2001:db8:1::1           20             0 100 ?
* i2001:db8:12::/64 2001:db8:3::1           20    100      0 100 ?
*>                  2001:db8:1::1           20             0 100 ?
* i2001:db8:13::/64 2001:db8:3::1           20    100      0 100 ?
*>                  2001:db8:1::1           20             0 100 ?

Number of prefixes 3
```

View of BGP_SW3's BGP table prior to Local Preference adjustment

BGP_SW3 currently uses the eBGP path via the directly connected (metered) link to AS 100, because the Local Preference values are the same. Because the Local Preference values are the same, the tie breaker is BGP's preference for the eBGP path over the iBGP path.

We want to make BGP_SW3 take the internal path via iBGP neighbor 2001:db8:2: :1.

```
BGP_SW3#show bgp ipv6
BGP table version is 5, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network             Next Hop            Metric LocPrf Weight Path
*> 2001:db8:11::/64 2001:db8:1::1           20             0 100 ?
*i                  2001:db8:2::1           20    100      0 100 ?
*> 2001:db8:12::/64 2001:db8:1::1           20             0 100 ?
*i                  2001:db8:2::1           20    100      0 100 ?
*> 2001:db8:13::/64 2001:db8:1::1           20             0 100 ?
*i                  2001:db8:2::1           20    100      0 100 ?
Number of prefixes 3
```

1. Create an Access List or a Prefix List to classify prefix 2001:db8:12::/64.

Option 1:

```
BGP_SW2(config)#ipv6 prefix-list REMOTE_NETWORK_LIST seq 10 permit
   2001:db8:11::/64
BGP_SW2(config)#ipv6 prefix-list REMOTE_NETWORK_LIST seq 20 permit
   2001:db8:12::/64
BGP_SW2(config)#ipv6 prefix-list REMOTE_NETWORK_LIST seq 30 permit
   2001:db8:13::/64
```

Option 2:

```
BGP_SW2(config)#ipv6 access-list standard REMOTE_NETWORK_LIST permit
   2001:db8:11::/64
BGP_SW2(config)#ipv6 access-list standard REMOTE_NETWORK_LIST permit
   2001:db8:12::/64
BGP_SW2(config)#ipv6 access-list standard REMOTE_NETWORK_LIST permit
   2001:db8:13::/64
```

2. Create a Route Map to match on the ACL or Prefix List and apply a weight value for the matching prefix(es).

- Sequence 10 will apply a Local Preference value of 150 to prefixes matching the ACL / Prefix List.
- Sequence 20 will permit all other prefixes, leaving the default value unchanged.

Option 1: (ACL used)

```
BGP_SW2(config)#route-map REMOTE_NETWORK permit 10
BGP_SW2(config-route-map)#match ipv6 address REMOTE_NETWORK_LIST
BGP_SW2(config-route-map)#set local-preference 150
BGP_SW2(config-route-map)#route-map REMOTE_NETWORK permit 20
```

Option 2: (Prefix List used)

```
BGP_SW2(config)#route-map REMOTE_NETWORK permit 10
BGP_SW2(config-route-map)#match ipv6 address prefix-list
   REMOTE_NETWORK_LIST
BGP_SW2(config-route-map)#set local-preference 150
BGP_SW2(config-route-map)#route-map REMOTE_NETWORK permit 20
```

3. Apply Route Map for prefixes advertised to iBGP neighbor BGP_SW3 in AS 200.

Route map will apply for prefixes advertised to iBGP neighbor 2001:db8:3::1.

```
BGP_SW2(config)#router bgp 200
BGP_SW2(config-router)#address-family ipv6
BGP_SW2(config-router-af)#neighbor 2001:db8:3::1 route-map
   REMOTE_NETWORK out
```

4. Clear BGP session to iBGP neighbor 2001:db8:3::1 (BGP_SW3) to apply the change.

```
BGP_SW2#clear bgp ipv6 2001:db8:3::1
```

Verify using the:

- BGP table seen on BGP_SW3. Prefixes advertised from BGP_SW2 have a local preference value of 150.
- Routing table seen on BGP_SW3.
- BGP neighbor 2001:db8:3::1 output seen on BGP_SW2 (Next Hop, Outbound path policy configured, Route map for outgoing advertisements).
- show route-map command on BGP_SW2.
- show ipv6 prefix-list detail command on BGP_SW2.

```
BGP_SW3#show bgp ipv6
BGP table version is 8, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*>i2001:db8:11::/64 2001:db8:2::1           20    150      0 100 ?
*                   2001:db8:1::1           20             0 100 ?
*>i2001:db8:12::/64 2001:db8:2::1           20    150      0 100 ?
*                   2001:db8:1::1           20             0 100 ?
*>i2001:db8:13::/64 2001:db8:2::1           20    150      0 100 ?
*                   2001:db8:1::1           20             0 100 ?

Number of prefixes 3
```

```
BGP_SW3#show ipv6 route database
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
       IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
       > - selected route, * - FIB route, p - stale info
Timers: Uptime

S  *> 2001:db8:1::1/128 [1/0] via fe80::200:cdff:fe2a:eaf1, vlan30, 03:19:24
S  *> 2001:db8:2::1/128 [1/0] via fe80::209:41ff:fefb:c323, vlan40, 03:19:46
C  *> 2001:db8:3::/64 via ::, lo, 03:28:00
B  *> 2001:db8:11::/64 [200/20] via 2001:db8:2::1 (recursive via fe80::209:41ff:fefb:c323),
vlan40, 00:11:29
B  *> 2001:db8:12::/64 [200/20] via 2001:db8:2::1 (recursive via fe80::209:41ff:fefb:c323),
vlan40, 00:11:29
B  *> 2001:db8:13::/64 [200/20] via 2001:db8:2::1 (recursive via fe80::209:41ff:fefb:c323),
vlan40, 00:11:29
C  *> 2001:db8:30::/64 via ::, vlan30, 03:27:43
C  *> 2001:db8:40::/64 via ::, vlan40, 03:05:08
C  *  fe80::/64 via ::, vlan40, 03:05:08
C  *> fe80::/64 via ::, lo, 03:32:52
C  *  fe80::/64 via ::, vlan30, 03:36:33
```

```
BGP_SW2#show bgp ipv6 neighbors 2001:db8:3::1
BGP neighbor is 2001:db8:3::1, remote AS 200, local AS 200, internal link
  BGP version 4, remote router ID 3.3.3.3
  BGP state = Established, up for 00:12:28
  Last read 00:12:28, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    4-Octet ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 247 messages, 5 notifications, 0 in queue
  Sent 254 messages, 8 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 5 seconds
  Update source is lo
 For address family: IPv6 Unicast
  BGP table version 15, neighbor version 15
  Index 0, Offset 0, Mask 0x1
  NEXT_HOP is always this router
  Community attribute sent to this neighbor (both)
  Outbound path policy configured
  Route map for outgoing advertisements is *REMOTE_NETWORK
  0 accepted prefixes
  3 announced prefixes

 Connections established 13; dropped 12
Local host: 2001:db8:2::1, Local port: 37155
Foreign host: 2001:db8:3::1, Foreign port: 179
Nexthop: 2.2.2.2
Nexthop global: 2001:db8:2::1
Nexthop local: ::
BGP connection: non shared network
Last Reset: 00:12:29, due to BGP Notification sent
Notification Error Message: (Cease/Other Configuration Change.)
```

```
BGP_SW2#show route-map
route-map REMOTE_NETWORK, permit, sequence 10
  Match clauses:
    ipv6 address prefix-list REMOTE_NETWORK_LIST
  Set clauses:
    local-preference 150
route-map REMOTE_NETWORK, permit, sequence 20
  Match clauses:
  Set clauses:
```

```
BGP_SW2#show ipv6 prefix-list detail
Prefix-list with the last deletion/insertion: REMOTE_NETWORK_LIST
ipv6 prefix-list REMOTE_NETWORK_LIST:
   count: 3, range entries: 0, sequences: 10 - 30
    ripd:
     seq 10 permit 2001:db8:11::/64 (hit count: 0, refcount: 0)
     seq 20 permit 2001:db8:12::/64 (hit count: 0, refcount: 0)
     seq 30 permit 2001:db8:13::/64 (hit count: 0, refcount: 0)
    ripngd:
     seq 10 permit 2001:db8:11::/64 (hit count: 0, refcount: 0)
     seq 20 permit 2001:db8:12::/64 (hit count: 0, refcount: 0)
     seq 30 permit 2001:db8:13::/64 (hit count: 0, refcount: 0)
    ospfd:
     seq 10 permit 2001:db8:11::/64 (hit count: 0, refcount: 0)
     seq 20 permit 2001:db8:12::/64 (hit count: 0, refcount: 0)
     seq 30 permit 2001:db8:13::/64 (hit count: 0, refcount: 0)
    ospf6d:
     seq 10 permit 2001:db8:11::/64 (hit count: 0, refcount: 0)
     seq 20 permit 2001:db8:12::/64 (hit count: 0, refcount: 0)
     seq 30 permit 2001:db8:13::/64 (hit count: 0, refcount: 0)
    bgpd:
     seq 10 permit 2001:db8:11::/64 (hit count: 1, refcount: 30)
     seq 20 permit 2001:db8:12::/64 (hit count: 1, refcount: 20)
     seq 30 permit 2001:db8:13::/64 (hit count: 1, refcount: 10)
```

BGP_SW2 configuration

```
BGP_SW2#
!
hostname BGP_SW2
!
vlan database
 vlan 20,40 state enable
!
interface port1.0.1
 switchport access vlan 20
!
interface port1.0.2
 switchport access vlan 40
!
interface lo
 ipv6 address 2001:db8:2::1/64
!
interface vlan20
 ipv6 address 2001:db8:20::2/64
!
interface vlan40
 ipv6 address 2001:db8:40::1/64
!
router bgp 200
 bgp router-id 2.2.2.2
 no bgp default ipv4-unicast
 neighbor 2001:db8:1::1 remote-as 100
 neighbor 2001:db8:1::1 ebgp-multihop 2
 neighbor 2001:db8:1::1 update-source lo
 neighbor 2001:db8:3::1 remote-as 200
 neighbor 2001:db8:3::1 update-source lo
 !
 address-family ipv6
 neighbor 2001:db8:1::1 activate
 neighbor 2001:db8:3::1 activate
 neighbor 2001:db8:3::1 next-hop-self
 neighbor 2001:db8:3::1 route-map REMOTE_NETWORK out
 exit-address-family
!
ipv6 route 2001:db8:1::1/128 fe80::200:cdff:fe2a:eaf1 vlan20
ipv6 route 2001:db8:3::1/128 fe80::200:cdff:fe2a:e635 vlan40
!
ipv6 prefix-list REMOTE_NETWORK_LIST seq 10 permit 2001:db8:11::/64
ipv6 prefix-list REMOTE_NETWORK_LIST seq 20 permit 2001:db8:12::/64
ipv6 prefix-list REMOTE_NETWORK_LIST seq 30 permit 2001:db8:13::/64
!
route-map REMOTE_NETWORK permit 10
 match ipv6 address prefix-list REMOTE_NETWORK_LIST
 set local-preference 150
!
route-map REMOTE_NETWORK permit 20
!
end
```

BGP_SW3 configuration

```
BGP_SW3#
!
hostname BGP_SW3
!
vlan database
 vlan 30,40 state enable
!
interface port1.0.1
 switchport access vlan 30
!
interface port1.0.2
 switchport access vlan 40
!
interface lo
 ipv6 address 2001:db8:3::1/64
!
interface vlan30
 ipv6 address 2001:db8:30::2/64
!
interface vlan40
 ipv6 address 2001:db8:40::2/64
!
router bgp 200
 bgp router-id 3.3.3.3
 no bgp default ipv4-unicast
 neighbor 2001:db8:1::1 remote-as 100
 neighbor 2001:db8:1::1 ebgp-multihop 2
 neighbor 2001:db8:1::1 update-source lo
 neighbor 2001:db8:2::1 remote-as 200
 neighbor 2001:db8:2::1 update-source lo
 !
 address-family ipv6
 neighbor 2001:db8:1::1 activate
 neighbor 2001:db8:2::1 activate
 neighbor 2001:db8:2::1 next-hop-self
 exit-address-family
!
ipv6 route 2001:db8:1::1/128 fe80::200:cdff:fe2a:eaf1 vlan30
ipv6 route 2001:db8:2::1/128 fe80::209:41ff:fefb:c323 vlan40
!
end
```

Multi-Exit Discriminator (MED)

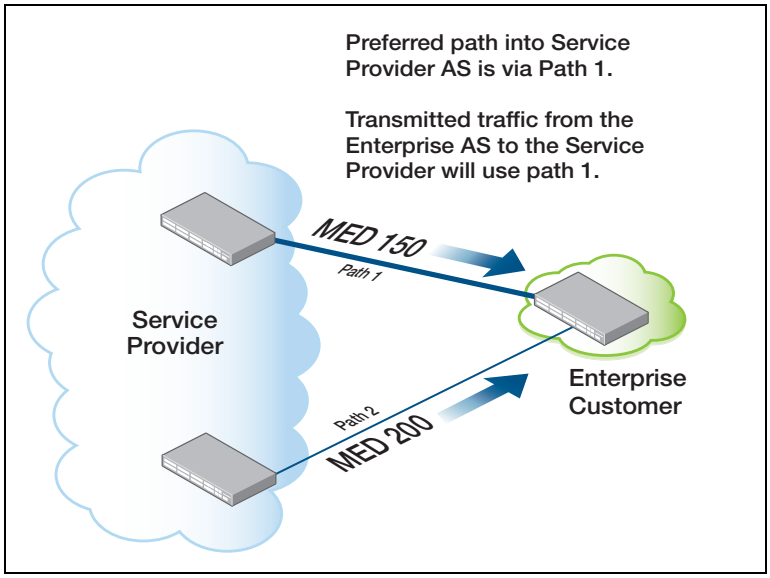
When to use MED

MED can be used to influence the best path when there are multiple entry points into an AS.

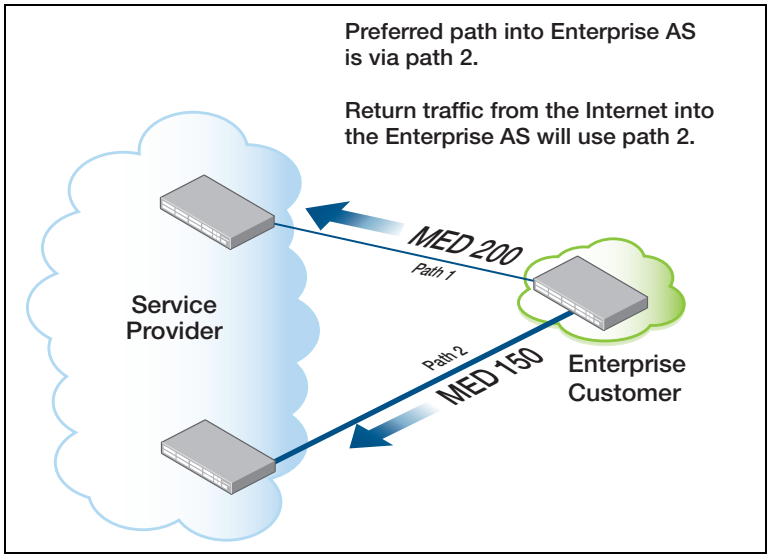
MED deployments

MED can be implemented in a customer network with multiple access points, and can be advertised to a neighboring AS to hint of the preferred path into the AS. The MED is also known as the external metric of a route. A lower MED value is preferred over a higher MED value.

For example, a provider may advertise MED to a customer AS to influence the customers best path into the provider's AS.



Another example is a customer AS may advertise MED to a provider's AS to influence the best path to destination networks residing in the customers AS.



Metric

Metric is used to determine the best path to a neighboring AS. Lower metric is preferred over a higher metric.

Design considerations

MED does not force the neighboring AS to send traffic back to the originating AS via the preferred path. MED only hints at the preferred path. The local policies of the neighboring AS BGP routers will determine if the MED will be enacted upon in the best path calculation.

This section also discusses several possibilities of MED not being compared.

Scenario 1 The neighbor AS removes the MED attribute upon receipt of the UPDATE message containing the attribute for the given advertised prefix. Neighboring AS BGP routers will not compare the MED advertised by the originating AS.

Neighbor AS configuration example

```
awplus(config)#router bgp 100
awplus(config-router)#bgp bestpath med remove-recv-med
```

Scenario 2 Due to the order of which the paths are learned from BGP peers residing in the same neighboring AS, the MED is not compared and the best path from that AS is not selected based on metric.

Mitigation of the issue

```
awplus(config)#router bgp 100
awplus(config-router)#bgp deterministic-med
```

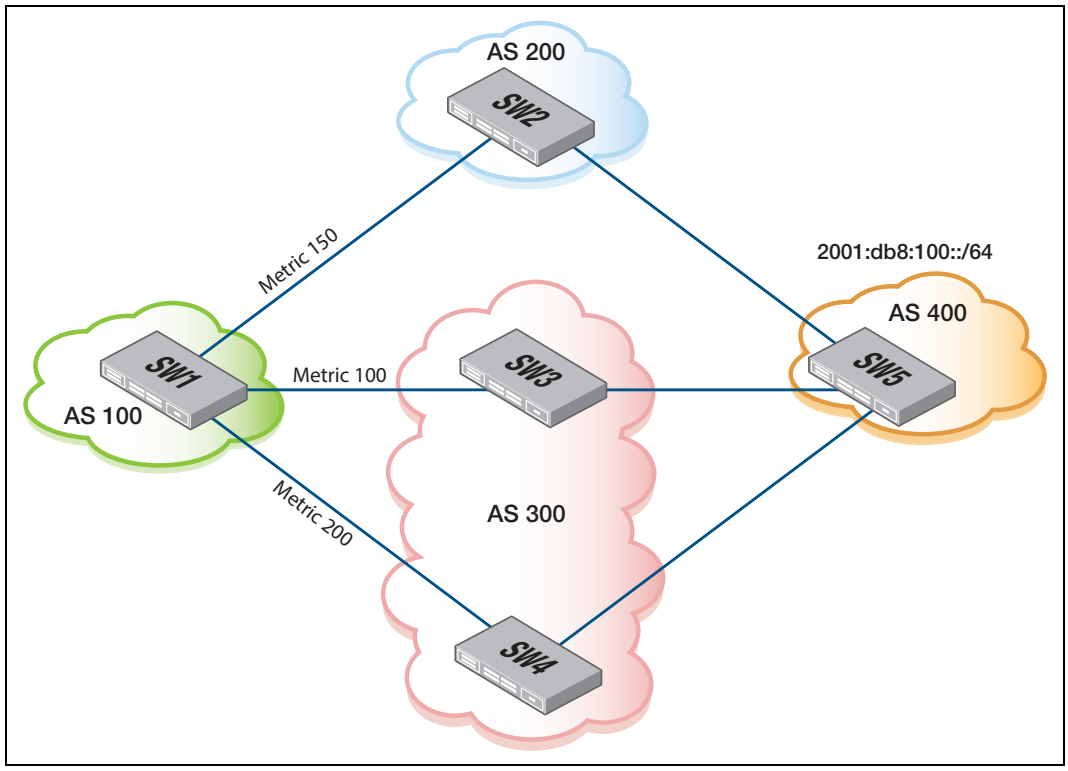
Scenario 3 MED is received from several peers which reside in different AS. Because MED is not compared between different AS by default, the best path is not selected based on metric.

Mitigation of the issue

```
awplus(config)#router bgp 100
awplus(config-router)#bgp always-compare-med
```

Using deterministic-med to ensure MED comparison from the same AS

Scenario



- SW1 in AS 100 learns the prefix 2001:db8:100::/64 from three external peers.
- SW1 must compare the MED from AS 300 external peers to select the best path out of the two.

BGP algorithm problem

- The BGP algorithm can lead to MED not being compared where two routes have an identical AS path to reach a prefix.
- This problem arises when two routes with an identical AS path advertised with MED, are not learned in order.
- BGP's best path algorithm evaluates the most recent path against the path below.
- If the routes are not learned in order, the best path algorithm will not compare the MED, as BGP requires the two routes to be from the same AS in order to conduct the MED evaluation.

Use

BGP Deterministic MED groups routes with an identical AS path regardless of the order learned, ensuring a MED comparison.

Implementation example

An AS is receiving routes with MED values from several AS. The AS is dual homed to one or more AS, and must evaluate the received routes MED from the same AS to find the best path from the advertising AS.

Objective

Set **BGP deterministic-med** on SW1 to compare the MED attribute for routes received from BGP peers in AS 300.

1. Create a route map to set a higher metric for the next preferred path.

```
SW2(config)#route-map SET_METRIC permit 10
SW2(config-route-map)#set metric 150
SW2(config)#router bgp 200
SW2(config-router)#address-family ipv6
SW2(config-router-af)#neighbor 2001:db8:10::1 route-map SET_METRIC out
```

2. Create a route map to set the lowest metric for the most preferred path.

```
SW3(config)#route-map SET_METRIC permit 10
SW3(config-route-map)#set metric 100
SW3(config)#router bgp 300
SW3(config-router)#address-family ipv6
SW3(config-router-af)#neighbor 2001:db8:20::1 route-map SET_METRIC out
```

3. Create a route map to set the highest metric for the least preferred path.

```
SW4(config)#route-map SET_METRIC permit 10
SW4(config-route-map)#set metric 200
SW4(config)#router bgp 300
SW4(config-router)#address-family ipv6
SW4(config-router-af)#neighbor 2001:db8:30::1 route-map SET_METRIC out
```

4. Set BGP deterministic-med on SW1 to evaluate, and always choose the path with the lowest metric.

```
SW1(config)#route bgp 100
SW1(config-router)#bgp deterministic-med
```

The MED will be compared when the next BGP RIB scan is executed, which will re-run the best path algorithm, and the new best route will be selected. No session reset is required.

Example:

```
SW1#show bgp ipv6
BGP table version is 3, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*  2001:db8:100::/64
                    2001:db8:20::2(fe80::eecd:6dff:fe20:c23e)
                                           100             0 300 400 i
*>                  2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                           150             0 200 400 i
*                   2001:db8:30::2(fe80::eecd:6dff:fe48:e546)
                                           200             0 300 400 i

Number of prefixes 1
```

```
SW1#show bgp ipv6 2001:db8:100::/64
BGP routing table entry for 2001:db8:100::/64
Paths: (3 available, best #2, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
  2001:db8:20::2 2001:db8:30::2
  300 400
    2001:db8:20::2(fe80::eecd:6dff:fe20:c23e) from 2001:db8:20::2 (3.3.3.3)
    (fe80::eecd:6dff:fe20:c23e)
      Origin IGP metric 100, localpref 100, valid, external
      Last update: Wed Dec 11 21:45:26 2013
  200 400
    2001:db8:10::2(fe80::eecd:6dff:fe20:c26b) from 2001:db8:10::2 (2.2.2.2)
    (fe80::eecd:6dff:fe20:c26b)
      Origin IGP metric 150, localpref 100, valid, external, best
      Last update: Wed Dec 11 21:45:14 2013
  300 400
    2001:db8:30::2(fe80::eecd:6dff:fe48:e546) from 2001:db8:30::2 (4.4.4.4)
    (fe80::eecd:6dff:fe48:e546)
      Origin IGP metric 200, localpref 100, valid, external
      Last update: Wed Dec 11 21:45:09 2013
```

BGP configuration of SW1

```
SW1#show running-config bgp
!
router bgp 100
 bgp router-id 1.1.1.1
 no bgp default ipv4-unicast
 bgp deterministic-med
 neighbor 2001:db8:10::2 remote-as 200
 neighbor 2001:db8:20::2 remote-as 300
 neighbor 2001:db8:30::2 remote-as 400
 !
 address-family ipv6
 neighbor 2001:db8:10::2 activate
 neighbor 2001:db8:20::2 activate
 neighbor 2001:db8:30::2 activate
 exit-address-family
!
```

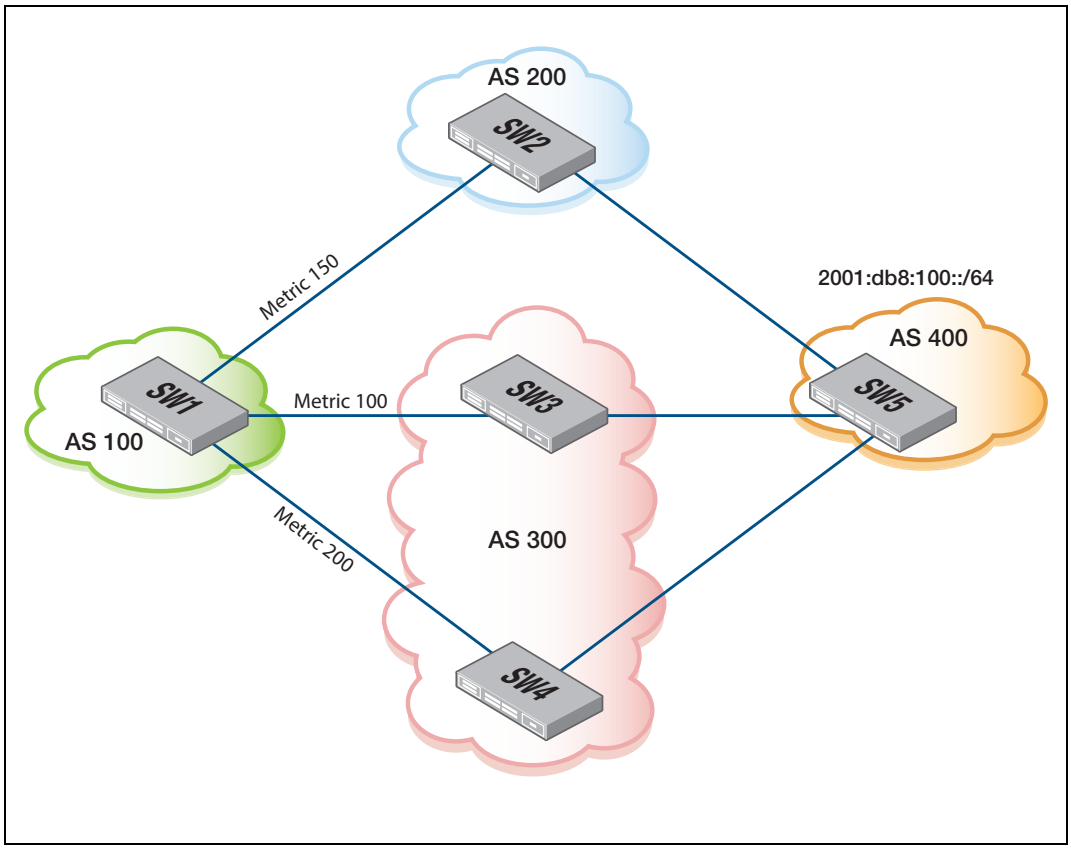
BGP configuration of SW2, SW3, SW4 and SW5

```
SW2#show running-config bgp
!
router bgp 200
 bgp router-id 2.2.2.2
 no bgp default ipv4-unicast
 neighbor 2001:db8:10::1 remote-as 100
 neighbor 2001:db8:40::2 remote-as 500
 !
 address-family ipv6
 neighbor 2001:db8:10::1 activate
 neighbor 2001:db8:10::1 route-map SET_METRIC out
 neighbor 2001:db8:40::2 activate
 exit-address-family
!
SW3#show running-config bgp
!
router bgp 300
 bgp router-id 3.3.3.3
 no bgp default ipv4-unicast
 neighbor 2001:db8:20::1 remote-as 100
 neighbor 2001:db8:50::2 remote-as 500
 !
 address-family ipv6
 neighbor 2001:db8:20::1 activate
 neighbor 2001:db8:20::1 route-map SET_METRIC out
 neighbor 2001:db8:50::2 activate
 exit-address-family
!
SW4#show running-config bgp
!
router bgp 400
 bgp router-id 4.4.4.4
 no bgp default ipv4-unicast
 neighbor 2001:db8:30::1 remote-as 100
 neighbor 2001:db8:60::2 remote-as 500
 !
 address-family ipv6
 neighbor 2001:db8:30::1 activate
 neighbor 2001:db8:30::1 route-map SET_METRIC out
 neighbor 2001:db8:60::2 activate
 exit-address-family
!
SW5#show running-config bgp
!
router bgp 500
 bgp router-id 5.5.5.5
 no bgp default ipv4-unicast
 neighbor 2001:db8:40::1 remote-as 200
 neighbor 2001:db8:50::1 remote-as 300
 neighbor 2001:db8:60::1 remote-as 400
 !
 address-family ipv6
 network 2001:db8:100::/64
 neighbor 2001:db8:40::1 activate
 neighbor 2001:db8:50::1 activate
 neighbor 2001:db8:60::1 activate
 exit-address-family
!
```

Comparing MED from different AS to change the path preference

Scenario

By default, BGP only compares MED for routes received from the same AS. In some cases, it is desired for MED to be compared for routes received from different AS.



Use

- The command bgp always-compare-med, requires BGP to compare the MED attribute when choosing paths advertised by different peers in different ASs.

Implementation example

- Multiple providers or enterprise customers agree on a policy for using MED.
- The enterprise customer wants to compare MED for routes received, to determine the best path to the destination.

Objective

Enable MED comparison for routes advertised by different AS.

View of BGP_SW1's BGP table prior to MED adjustment

- The current preferred path is via SW3 in AS 300 because it is the oldest path.
- The oldest path is seen by viewing the specific prefix.

```
SW1#show bgp ipv6
BGP table version is 2, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*  2001:db8:100::/64
                    2001:db8:20::2(fe80::eecd:6dff:fe20:c23e)
                                           100             0 300 400 i
*                   2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                           150             0 200 400 i
*>                  2001:db8:30::2(fe80::eecd:6dff:fe48:e546)
                                           200             0 300 400 i

Number of prefixes 1
```

```
SW1#show bgp ipv6 2001:db8:100::/64
BGP routing table entry for 2001:db8:100::/64
Paths: (3 available, best #3, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
  2001:db8:10::2 2001:db8:20::2
  300 400
    2001:db8:20::2(fe80::eecd:6dff:fe20:c23e) from 2001:db8:20::2 (3.3.3.3)
    (fe80::eecd:6dff:fe20:c23e)
      Origin IGP metric 100, localpref 100, valid, external
      Last update: Wed Dec 11 21:45:26 2013
  200 400
    2001:db8:10::2(fe80::eecd:6dff:fe20:c26b) from 2001:db8:10::2 (2.2.2.2)
    (fe80::eecd:6dff:fe20:c26b)
      Origin IGP metric 150, localpref 100, valid, external
      Last update: Wed Dec 11 21:45:14 2013
  300 400
    2001:db8:30::2(fe80::eecd:6dff:fe48:e546) from 2001:db8:30::2 (4.4.4.4)
    (fe80::eecd:6dff:fe48:e546)
      Origin IGP metric 200, localpref 100, valid, external, best
      Last update: Wed Dec 11 21:45:09 2013
```

1. Create a route map to set the lowest metric for the next preferred path.

```
SW2(config)#route-map SET_METRIC permit 10
SW2(config-route-map)#set metric 100
SW2(config)#router bgp 200
SW2(config-router)#address-family ipv6
SW2(config-router-af)#neighbor 2001:db8:10::1 route-map SET_METRIC out
```

2. Create a route map to set a lowest metric for the most preferred path.

```
SW3(config)#route-map SET_METRIC permit 10
SW3(config-route-map)#set metric 150
SW3(config)#router bgp 300
SW3(config-router)#address-family ipv6
SW3(config-router-af)#neighbor 2001:db8:20::1 route-map SET_METRIC out
```

3. Create a route map to set the highest metric for the least preferred path.

```
SW4(config)#route-map SET_METRIC permit 10
SW4(config-route-map)#set metric 200
SW4(config)#router bgp 400
SW4(config-router)#address-family ipv6
SW4(config-router-af)#neighbor 2001:db8:30::1 route-map SET_METRIC out
```

4. Set SW1 to compare MED values from different AS in order to select the best path based on lowest metric.

```
SW1(config)#router bgp 100
SW1(config-route-map)#bgp always-compare-med
```

The MED will be compared when the next BGP RIB scan is executed, which will re-run the best path algorithm, and the new best route will be selected. No session reset is required.

Verify by viewing the:

- BGP table. We can see SW1 has learned the destination network from the three external BGP peers.
- Specific BGP table prefix entry.

```
SW1#show bgp ipv6
BGP table version is 2, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*  2001:db8:100::/64
                    2001:db8:30::2(fe80::eecd:6dff:fe48:e546)
                                           200             0 300 400 i
*>                  2001:db8:20::2(fe80::eecd:6dff:fe20:c23e)
                                           100             0 300 400 i
*                   2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                           150             0 200 400 i

Number of prefixes 1
```

```
SW1#show bgp ipv6 2001:db8:100::/64
BGP routing table entry for 2001:db8:100::/64
Paths: (3 available, best #2, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
  2001:db8:10::2 2001:db8:30::2
  300 400
    2001:db8:30::2(fe80::eecd:6dff:fe48:e546) from 2001:db8:30::2 (4.4.4.4)
    (fe80::eecd:6dff:fe48:e546)
      Origin IGP metric 200, localpref 100, valid, external
      Last update: Thu Dec 12 01:41:36 2013
  300 400
    2001:db8:20::2(fe80::eecd:6dff:fe20:c23e) from 2001:db8:20::2 (3.3.3.3)
    (fe80::eecd:6dff:fe20:c23e)
      Origin IGP metric 100, localpref 100, valid, external, best
      Last update: Thu Dec 12 01:41:35 2013
  200 400
    2001:db8:10::2(fe80::eecd:6dff:fe20:c26b) from 2001:db8:10::2 (2.2.2.2)
    (fe80::eecd:6dff:fe20:c26b)
      Origin IGP metric 150, localpref 100, valid, external
      Last update: Thu Dec 12 01:41:30 2013
```

BGP configuration of SW1

```
SW1#show run bgp
!
router bgp 100
 bgp router-id 1.1.1.1
 bgp always-compare-med
 no bgp default ipv4-unicast
 bgp deterministic-med
 neighbor 2001:db8:10::2 remote-as 200
 neighbor 2001:db8:20::2 remote-as 300
 neighbor 2001:db8:30::2 remote-as 300
!
 address-family ipv6
 neighbor 2001:db8:10::2 activate
 neighbor 2001:db8:20::2 activate
 neighbor 2001:db8:30::2 activate
 exit-address-family
!
```

BGP configuration of SW2

```
SW2#show running-config bgp
!
router bgp 200
 bgp router-id 2.2.2.2
 no bgp default ipv4-unicast
 neighbor 2001:db8:10::1 remote-as 100
 neighbor 2001:db8:40::2 remote-as 500
 !
 address-family ipv6
 neighbor 2001:db8:10::1 activate
 neighbor 2001:db8:10::1 route-map SET_METRIC out
 neighbor 2001:db8:40::2 activate
 exit-address-family
!
```

BGP configuration of SW3

```
SW3#show running-config bgp
!
router bgp 300
 bgp router-id 3.3.3.3
 no bgp default ipv4-unicast
 neighbor 2001:db8:20::1 remote-as 100
 neighbor 2001:db8:50::2 remote-as 500
 !
 address-family ipv6
 neighbor 2001:db8:20::1 activate
 neighbor 2001:db8:20::1 route-map SET_METRIC out
 neighbor 2001:db8:50::2 activate
 exit-address-family
!
```

BGP configuration of SW4

```
SW4#show running-config bgp
!
router bgp 400
 bgp router-id 4.4.4.4
 no bgp default ipv4-unicast
 neighbor 2001:db8:30::1 remote-as 100
 neighbor 2001:db8:60::2 remote-as 500
 !
 address-family ipv6
 neighbor 2001:db8:30::1 activate
 neighbor 2001:db8:30::1 route-map SET_METRIC out
 neighbor 2001:db8:60::2 activate
 exit-address-family
!
```

BGP configuration of SW5

```
SW5#show running-config bgp
!
router bgp 500
 bgp router-id 5.5.5.5
 no bgp default ipv4-unicast
 neighbor 2001:db8:40::1 remote-as 200
 neighbor 2001:db8:50::1 remote-as 300
 neighbor 2001:db8:60::1 remote-as 400
 !
 address-family ipv6
 network 2001:db8:100::/64
 neighbor 2001:db8:40::1 activate
 neighbor 2001:db8:50::1 activate
 neighbor 2001:db8:60::1 activate
 exit-address-family
!
```

Using deterministic-med and always-compare-med together to change the path preference

The process overview explained

Initially the paths to destination network 2001:db8:100::/64 are not learned in an order which permits MED being compared.

The BGP best path algorithm compares the first path to the next path. Because MED is not compared, the BGP algorithm moves to the next preference to determine the best path.

Before bgp deterministic-med is enabled

Best path is currently the highest metric path via AS 300 based on the path being the oldest entry.

```
1: ASPATH 300, MED 200, external
2: ASPATH 200, MED 150, external
3: ASPATH 300, MED 100, external
```

BGP deterministic-med is then enabled and when the next BGP RIB scan occurs, the best path algorithm is re-run. This time however, entries are sorted by ASN so the metric can be compared.

The paths from Group 1 are compared based on lowest metric. The best path from Group 1 is the path via AS 300 with the lower metric of 100.

This best path from Group 1 is then compared to the path in AS 200. Because the metric is not compared between different ASNs by default, the preference then falls to path age. The path via AS 200 is the oldest path out of the two and thus becomes the best route.

After bgp deterministic-med is enabled: (and before always-compare-med is enabled)

The best path is now via AS 200 based on the path being the oldest entry when compared to Group 1's best selected path with MED of 100).

```
Group 1:
1: ASPATH 300, MED 100, external, most recent
3: ASPATH 300, MED 200, external, oldest
Group 2:
2: ASPATH 200, MED 150, external, next oldest
```

BGP always-compare-med is then enabled and when the next BGP RIP scan occurs, the best path algorithm is re-run. The entries grouped using deterministic-med are compared and the best path from AS 300 is determined as being the route with a metric of 100. This time however when the path is compared to the path in Group 2, the metric is compared. The best path is now the lowest metric path of 100 via AS 300 to reach the destination.

When bgp always-compare-med is enabled:

Best path is now via the AS 300 path with a MED of 100, due to the path having the lowest MED out of all paths.

```
Group 1:
1: ASPATH 300, MED 100, external, most recent
3: ASPATH 300, MED 200, external, oldest
Group 2:
2: ASPATH 200, MED 150, external, next oldest
```

Example

Part 1: When BGP deterministic-med and always-compare-med is off

Note: Always-compare-med is also turned off so MED will not be compared between different AS, it will only be compared from the same AS.

In this BGP table output following, due to the order in which the prefix was installed, the current best path selected to the 2001:db8:100::/64 prefix uses the highest metric path. This is due to both paths learned via AS 300 were not learned in order for which they could be compared.

Because the metric was not compared, the path via peer 2001:db8:30::2 is the oldest route and is preferred over the other two routes.

```
SW1#show bgp ipv6
BGP table version is 2, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*  2001:db8:100::/64
                    2001:db8:20::2(fe80::eecd:6dff:fe20:c23e)
                                          100             0 300 400 i
*                   2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                          150             0 200 400 i
*>                  2001:db8:30::2(fe80::eecd:6dff:fe48:e546)
                                          200             0 300 400 i

Number of prefixes 1
```

- The path via peer 2001:db8:30::2 in AS 300 was learned first.
- Then the path via peer 2001:db8:10::2 in AS 200 was learned.
- Finally the path via peer 2001:db8:20::2 in AS 300 was learned.

```
SW1#show bgp ipv6 2001:db8:100::/64
BGP routing table entry for 2001:db8:100::/64
Paths: (3 available, best #3, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
  2001:db8:10::2 2001:db8:20::2
  300 400
    2001:db8:20::2(fe80::eecd:6dff:fe20:c23e) from 2001:db8:20::2 (3.3.3.3)
    (fe80::eecd:6dff:fe20:c23e)
      Origin IGP metric 100, localpref 100, valid, external
      Last update: Wed Dec 11 21:45:26 2013
  200 400
    2001:db8:10::2(fe80::eecd:6dff:fe20:c26b) from 2001:db8:10::2 (2.2.2.2)
    (fe80::eecd:6dff:fe20:c26b)
      Origin IGP metric 150, localpref 100, valid, external
      Last update: Wed Dec 11 21:45:14 2013
  300 400
    2001:db8:30::2(fe80::eecd:6dff:fe48:e546) from 2001:db8:30::2 (4.4.4.4)
    (fe80::eecd:6dff:fe48:e546)
      Origin IGP metric 200, localpref 100, valid, external, best
      Last update: Wed Dec 11 21:45:09 2013
```

Part 2: Ensuring a comparison of MED from the same AS:

```
SW1(config)#router bgp 100
SW1(config-router)#bgp deterministic-med
SW1(config-router)#end
```

BGP executes a scan of the BGP routing table (Loc-RIB) during the next interval.

```
22:02:35 SW1 BGP[1499]: [RIB] Scanning BGP RIB...
```

The best path is now deemed to be via BGP peer 2001:db8:10::2 in AS 200.

MED has been compared between the two paths from AS 300 and the best route from AS 300 is via the peer 2001:db8:20::1 with a lower metric of 100.

However, this path is then compared against the path via AS 200. Because always-compare-med is not enabled by default, the next preference again moves to age comparison.

The path via SW2 is the best path because it is older than the path via SW3 in AS 300.

Before BGP always-compare-med is enabled

- As seen in the output below, BGP deterministic-med has been enabled, but always-compare-med has not.

```
SW1#show running-config bgp
!
router bgp 100
 bgp router-id 1.1.1.1
 no bgp default ipv4-unicast
 bgp deterministic-med
 neighbor 2001:db8:10::2 remote-as 200
 neighbor 2001:db8:20::2 remote-as 300
 neighbor 2001:db8:30::2 remote-as 300
 !
 address-family ipv6
 neighbor 2001:db8:10::2 activate
 neighbor 2001:db8:20::2 activate
 neighbor 2001:db8:30::2 activate
 exit-address-family
!
```

- Currently the best path is selected based on oldest route.

```
SW1#show bgp ipv6
BGP table version is 3, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*  2001:db8:100::/64
                    2001:db8:20::2(fe80::eecd:6dff:fe20:c23e)
                                             100             0 300 400 i
*>                  2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                             150             0 200 400 i
*                   2001:db8:30::2(fe80::eecd:6dff:fe48:e546)
                                             200             0 300 400 i

Number of prefixes 1
```

- The current best path is via the BGP peer 2001:db8:20::2, however the desired path is the path with the lowest metric. To achieve this we need to configure bgp always-compare-med, to ensure the comparison of MED between different AS.

```
SW1#show bgp ipv6 2001:db8:100::/64
BGP routing table entry for 2001:db8:100::/64
Paths: (3 available, best #2, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
  2001:db8:20::2 2001:db8:30::2
  300 400
    2001:db8:20::2(fe80::eecd:6dff:fe20:c23e) from 2001:db8:20::2 (3.3.3.3)
    (fe80::eecd:6dff:fe20:c23e)
      Origin IGP metric 100, localpref 100, valid, external
      Last update: Wed Dec 11 21:45:26 2013
  200 400
    2001:db8:10::2(fe80::eecd:6dff:fe20:c26b) from 2001:db8:10::2 (2.2.2.2)
    (fe80::eecd:6dff:fe20:c26b)
      Origin IGP metric 150, localpref 100, valid, external, best
      Last update: Wed Dec 11 21:45:14 2013
  300 400
    2001:db8:30::2(fe80::eecd:6dff:fe48:e546) from 2001:db8:30::2 (4.4.4.4)
    (fe80::eecd:6dff:fe48:e546)
      Origin IGP metric 200, localpref 100, valid, external
      Last update: Wed Dec 11 21:45:09 2013
```

Part 3: When always-compare-med is enabled

```
SW1(config-router)#bgp always-compare-med
```

BGP executes a scan of the BGP routing table (Loc-RIB) during the next interval and re-runs the best path algorithm.

```
22:29:35 SW1 BGP[1499]: [RIB] Scanning BGP RIB...
```

The best path to the destination is via peer 2001:db8:20::2 in AS 300. This result has occurred due to the metric being compared between the routes from each AS.

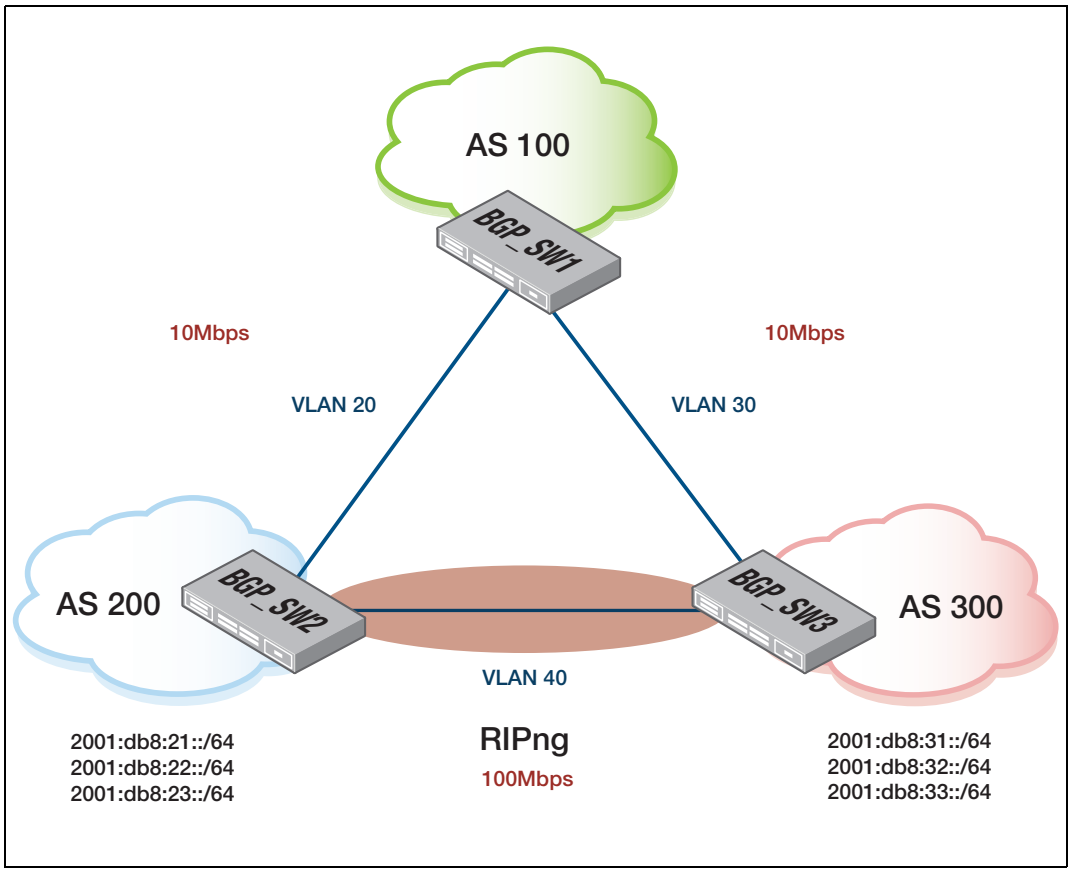
Because this path has the lowest metric, it is the best path.

```
SW1#show bgp ipv6
BGP table version is 4, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:100::/64
                    2001:db8:20::2(fe80::eecd:6dff:fe20:c23e)
                                             100             0 300 400 i
*                   2001:db8:10::2(fe80::eecd:6dff:fe20:c26b)
                                             150             0 200 400 i
*                   2001:db8:30::2(fe80::eecd:6dff:fe48:e546)
                                             200             0 300 400 i

Number of prefixes 1
```

```
SW1#show bgp ipv6 2001:db8:100::/64
BGP routing table entry for 2001:db8:100::/64
Paths: (3 available, best #1, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
  2001:db8:10::2 2001:db8:30::2
  300 400
    2001:db8:20::2(fe80::eecd:6dff:fe20:c23e) from 2001:db8:20::2 (3.3.3.3)
    (fe80::eecd:6dff:fe20:c23e)
      Origin IGP metric 100, localpref 100, valid, external, best
      Last update: Wed Dec 11 21:45:26 2013
  200 400
    2001:db8:10::2(fe80::eecd:6dff:fe20:c26b) from 2001:db8:10::2 (2.2.2.2)
    (fe80::eecd:6dff:fe20:c26b)
      Origin IGP metric 150, localpref 100, valid, external
      Last update: Wed Dec 11 21:45:14 2013
  300 400
    2001:db8:30::2(fe80::eecd:6dff:fe48:e546) from 2001:db8:30::2 (4.4.4.4)
    (fe80::eecd:6dff:fe48:e546)
      Origin IGP metric 200, localpref 100, valid, external
      Last update: Wed Dec 11 21:45:09 2013
```

Using the BGP backdoor feature to change path preference without altering the default BGP AD

Scenario



Objective

A business acquisition has occurred and a new RIPng network is being established between AS 200 and AS 300.

- This new link is much faster with a data rate of 100Mbps.
- The older links which BGP uses have a data rate of only 10Mbps.
- Currently no traffic transits the new RIPng network. Traffic between AS200 and AS300 flows via AS100.
- This is because the default eBGP Administrative Distance is 20, and RIPng is 120. The lower Administrative Distance (AD) wins and the slower eBGP path is preferred.
- The backdoor feature (parameter) must be used to ensure traffic between AS 200 and AS 300 uses the faster RIPng network.
- Using the backdoor feature will avoid changing the default AD for all BGP paths.
- The eBGP path will become the backup path for the network if the RIPng path goes down.

View of BGP_SW2's BGP, RIPng and routing tables prior to the backdoor route being installed

All traffic to prefixes in AS 300 will be sent via the slower external path transiting AS 100, as the AD 20 is lower than RIPng's AD of 120. RIPng knows of the prefixes in AS 300.

```
BGP_SW2#show bgp ipv6
BGP table version is 16, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:1::1/128
                    fe80::200:cdff:fe2a:eaf1
                                                       32768 ?
*> 2001:db8:21::/64 ::                                 32768 ?
*> 2001:db8:22::/64 ::                                 32768 ?
*> 2001:db8:23::/64 ::                                 32768 ?
*> 2001:db8:31::/64 2001:db8:1::1           0              0 100 300  ?
*> 2001:db8:32::/64 2001:db8:1::1           0              0 100 300  ?
*> 2001:db8:33::/64 2001:db8:1::1           0              0 100 300  ?

Number of prefixes 7
```

```
BGP_SW2#show ipv6 rip database

Codes: R - RIP, Rc - RIP connected, Rs - RIP static, Ra - RIP aggregated,
       Rcx - RIP connect suppressed, Rsx - RIP static suppressed,
       C - Connected, S - Static, O - OSPF, B - BGP

   Network                    Next Hop                  If       Met Tag  Time
S  2001:db8:1::1/128          fe80::200:cdff:fe2a:eaf1  vlan20     1  0
S  2001:db8:21::/64           ::                         --        1  0
S  2001:db8:22::/64           ::                         --        1  0
S  2001:db8:23::/64           ::                         --        1  0
R  2001:db8:31::/64           fe80::200:cdff:fe2a:e635  vlan40     2  0 02:35
R  2001:db8:32::/64           fe80::200:cdff:fe2a:e635  vlan40     2  0 02:35
R  2001:db8:33::/64           fe80::200:cdff:fe2a:e635  vlan40     2  0 02:35
Rc 2001:db8:40::/64           ::                        vlan40     1  0
```

```
BGP_SW2#show ipv6 route database
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
       IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
       > - selected route, * - FIB route, p - stale info
Timers: Uptime

S  *> 2001:db8:1::1/128 [1/0] via fe80::200:cdff:fe2a:eaf1, vlan20, 00:24:31
C  *> 2001:db8:2::/64 via ::, lo, 00:24:31
C  *> 2001:db8:20::/64 via ::, vlan20, 00:24:24
S  *> 2001:db8:21::/64 [1/0] via ::, Null, 00:20:52
S  *> 2001:db8:22::/64 [1/0] via ::, Null, 00:20:50
S  *> 2001:db8:23::/64 [1/0] via ::, Null, 00:20:47
B  *> 2001:db8:31::/64 [20/0] via 2001:db8:1::1 (recursive via fe80::200:cdff:fe2a:eaf1), vlan20,
00:14:31
R     2001:db8:31::/64 [120/2] via fe80::200:cdff:fe2a:e635, vlan40, 00:15:22
B  *> 2001:db8:32::/64 [20/0] via 2001:db8:1::1 (recursive via fe80::200:cdff:fe2a:eaf1), vlan20,
00:14:31
R     2001:db8:32::/64 [120/2] via fe80::200:cdff:fe2a:e635, vlan40, 00:15:22
B  *> 2001:db8:33::/64 [20/0] via 2001:db8:1::1 (recursive via fe80::200:cdff:fe2a:eaf1), vlan20,
00:14:31
R     2001:db8:33::/64 [120/2] via fe80::200:cdff:fe2a:e635, vlan40, 00:15:22
R     2001:db8:40::/64 [0/1] via ::, vlan40, 00:23:03
C  *> 2001:db8:40::/64 via ::, vlan40, 00:24:23
C  *  fe80::/64 via ::, vlan40, 00:24:23
C  *  fe80::/64 via ::, vlan20, 00:24:24
C  *> fe80::/64 via ::, lo, 00:24:31
```

```
BGP_SW3#show bgp ipv6
BGP table version is 20, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:1::1/128
                    fe80::200:cdff:fe2a:eaf1
                                                       32768 ?
*> 2001:db8:21::/64 2001:db8:1::1            0             0 100 200 ?
*> 2001:db8:22::/64 2001:db8:1::1            0             0 100 200 ?
*> 2001:db8:23::/64 2001:db8:1::1            0             0 100 200 ?
*> 2001:db8:31::/64 ::                                    32768 ?
*> 2001:db8:32::/64 ::                                    32768 ?
*> 2001:db8:33::/64 ::                                    32768 ?
```

```
BGP_SW3#show ipv6 rip database

Codes: R - RIP, Rc - RIP connected, Rs - RIP static, Ra - RIP aggregated,
       Rcx - RIP connect suppressed, Rsx - RIP static suppressed,
       C - Connected, S - Static, O - OSPF, B - BGP

   Network                    Next Hop                  If       Met  Tag  Time
S  2001:db8:1::1/128          fe80::200:cdff:fe2a:eaf1  vlan30     1   0
R  2001:db8:21::/64           fe80::209:41ff:fefb:c323  vlan40     2   0 02:48
R  2001:db8:22::/64           fe80::209:41ff:fefb:c323  vlan40     2   0 02:48
R  2001:db8:23::/64           fe80::209:41ff:fefb:c323  vlan40     2   0 02:48
S  2001:db8:31::/64           ::                         --        1   0
S  2001:db8:32::/64           ::                         --        1   0
S  2001:db8:33::/64           ::                         --        1   0
Rc 2001:db8:40::/64           ::                        vlan40     1   0
```

```
BGP_SW3#show ipv6 route database
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
       IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
       > - selected route, * - FIB route, p - stale info
Timers: Uptime

S  *> 2001:db8:1::1/128 [1/0] via fe80::200:cdff:fe2a:eaf1, vlan30, 00:28:37
C  *> 2001:db8:3::/64 via ::, lo, 00:28:37
B  *> 2001:db8:21::/64 [20/0] via 2001:db8:1::1 (recursive via fe80::200:cdff:fe2a:eaf1), vlan30,
00:18:17
R     2001:db8:21::/64 [120/2] via fe80::209:41ff:fefb:c323, vlan40, 00:18:52
B  *> 2001:db8:22::/64 [20/0] via 2001:db8:1::1 (recursive via fe80::200:cdff:fe2a:eaf1), vlan30,
00:18:17
R     2001:db8:22::/64 [120/2] via fe80::209:41ff:fefb:c323, vlan40, 00:18:52
B  *> 2001:db8:23::/64 [20/0] via 2001:db8:1::1 (recursive via fe80::200:cdff:fe2a:eaf1), vlan30,
00:18:17
R     2001:db8:23::/64 [120/2] via fe80::209:41ff:fefb:c323, vlan40, 00:18:52
C  *> 2001:db8:30::/64 via ::, vlan30, 00:28:30
S  *> 2001:db8:31::/64 [1/0] via ::, Null, 00:23:57
S  *> 2001:db8:32::/64 [1/0] via ::, Null, 00:23:54
S  *> 2001:db8:33::/64 [1/0] via ::, Null, 00:23:51
R     2001:db8:40::/64 [0/1] via ::, vlan40, 00:24:11
C  *> 2001:db8:40::/64 via ::, vlan40, 00:28:30
C  *  fe80::/64 via ::, vlan40, 00:28:30
C  *  fe80::/64 via ::, vlan30, 00:28:30
C  *> fe80::/64 via ::, lo, 00:28:37
```

1. Configure the destination networks using the **backdoor** parameter.

- The backdoor parameter will set the specified network to an AD of 200 instead of 20.
- Since the AD is now higher, the IGP (RIPng) will be the preferred path.
- Only the BGP AD for the prefixes specified below will be modified. The default BGP AD remains unchanged for any other BGP prefixes.

```
BGP_SW2(config)#router bgp 200
BGP_SW2(config-router)#address-family ipv6
BGP_SW2(config-router-af)#network 2001:db8:31::/64 backdoor
BGP_SW2(config-router-af)#network 2001:db8:32::/64 backdoor
BGP_SW2(config-router-af)#network 2001:db8:33::/64 backdoor

BGP_SW3(config)#router bgp 300
BGP_SW3(config-router)#address-family ipv6
BGP_SW3(config-router-af)#network 2001:db8:21::/64 backdoor
BGP_SW3(config-router-af)#network 2001:db8:22::/64 backdoor
BGP_SW3(config-router-af)#network 2001:db8:23::/64 backdoor
```

Verify

- By viewing the routing table on BGP_SW2 and BGP_SW3. (Notice the eBGP paths now have an AD of 200.)
- The RIPng routes are now the best selected routes to reach the destination networks.

```
BGP_SW2#show ipv6 route database
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
       IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
       > - selected route, * - FIB route, p - stale info
Timers: Uptime

S  *> 2001:db8:1::1/128 [1/0] via fe80::200:cdff:fe2a:eaf1, vlan20, 00:34:01
C  *> 2001:db8:2::/64 via ::, lo, 00:34:01
C  *> 2001:db8:20::/64 via ::, vlan20, 00:33:53
S  *> 2001:db8:21::/64 [1/0] via ::, Null, 00:34:01
S  *> 2001:db8:22::/64 [1/0] via ::, Null, 00:34:01
S  *> 2001:db8:23::/64 [1/0] via ::, Null, 00:34:01
B     2001:db8:31::/64 [200/0] via 2001:db8:1::1, 00:24:45
R  *> 2001:db8:31::/64 [120/2] via fe80::200:cdff:fe2a:e635, vlan40, 00:27:59
B     2001:db8:32::/64 [200/0] via 2001:db8:1::1, 00:22:31
R  *> 2001:db8:32::/64 [120/2] via fe80::200:cdff:fe2a:e635, vlan40, 00:27:59
B     2001:db8:33::/64 [200/0] via 2001:db8:1::1, 00:22:28
R  *> 2001:db8:33::/64 [120/2] via fe80::200:cdff:fe2a:e635, vlan40, 00:27:59
R     2001:db8:40::/64 [0/1] via ::, vlan40, 00:28:31
C  *> 2001:db8:40::/64 via ::, vlan40, 00:28:31
C  *  fe80::/64 via ::, vlan40, 00:28:31
C  *  fe80::/64 via ::, vlan20, 00:33:53
C  *> fe80::/64 via ::, lo, 00:34:01
```

```
BGP_SW3#show ipv6 route database
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
       IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
       > - selected route, * - FIB route, p - stale info
Timers: Uptime

S  *> 2001:db8:1::1/128 [1/0] via fe80::200:cdff:fe2a:eaf1, vlan30, 00:28:49
C  *> 2001:db8:3::/64 via ::, lo, 00:28:49
B     2001:db8:21::/64 [200/0] via 2001:db8:1::1, 00:25:10
R  *> 2001:db8:21::/64 [120/2] via fe80::209:41ff:fefb:c323, vlan40, 00:28:37
B     2001:db8:22::/64 [200/0] via 2001:db8:1::1, 00:22:30
R  *> 2001:db8:22::/64 [120/2] via fe80::209:41ff:fefb:c323, vlan40, 00:28:37
B     2001:db8:23::/64 [200/0] via 2001:db8:1::1, 00:22:27
R  *> 2001:db8:23::/64 [120/2] via fe80::209:41ff:fefb:c323, vlan40, 00:28:37
C  *> 2001:db8:30::/64 via ::, vlan30, 00:28:42
S  *> 2001:db8:31::/64 [1/0] via ::, Null, 00:28:49
S  *> 2001:db8:32::/64 [1/0] via ::, Null, 00:28:49
S  *> 2001:db8:33::/64 [1/0] via ::, Null, 00:28:49
R     2001:db8:40::/64 [0/1] via ::, vlan40, 00:28:42
C  *> 2001:db8:40::/64 via ::, vlan40, 00:28:42
C  *  fe80::/64 via ::, vlan40, 00:28:42
C  *  fe80::/64 via ::, vlan30, 00:28:42
C  *> fe80::/64 via ::, lo, 00:28:49
```

Configuration of BGP_SW2

Note: Route filtering has been applied to BGP so the static routes to the neighbor's loopback is not advertised.

```
BGP_SW2#

!
hostname BGP_SW2
!
vlan database
 vlan 20,40 state enable
!
interface port1.0.1
 switchport access vlan 20
!
interface port1.0.2
 switchport access vlan 40
!
interface lo
 ipv6 address 2001:db8:2::1/64
!
interface vlan20
 ipv6 address 2001:db8:20::2/64
!
interface vlan40
 ipv6 address 2001:db8:40::1/64
 ipv6 router rip
!
router ipv6 rip
 redistribute static
 distribute-list prefix FILTER_OUT out
!
router bgp 200
 bgp router-id 2.2.2.2
 no bgp default ipv4-unicast
 neighbor 2001:db8:1::1 remote-as 100
 neighbor 2001:db8:1::1 ebgp-multihop 2
 neighbor 2001:db8:1::1 update-source lo
 !
 address-family ipv6
 network 2001:db8:31::/64 backdoor
 network 2001:db8:32::/64 backdoor
 network 2001:db8:33::/64 backdoor
 redistribute static
 neighbor 2001:db8:1::1 activate
 neighbor 2001:db8:1::1 prefix-list FILTER_OUT out
 exit-address-family
!
ipv6 route 2001:db8:1::1/128 fe80::200:cdff:fe2a:eaf1 vlan20
ipv6 route 2001:db8:21::/64 Null
ipv6 route 2001:db8:22::/64 Null
ipv6 route 2001:db8:23::/64 Null
!
ipv6 prefix-list FILTER_OUT seq 10 deny 2001:db8:1::1/128
ipv6 prefix-list FILTER_OUT seq 20 permit any
!
end
```

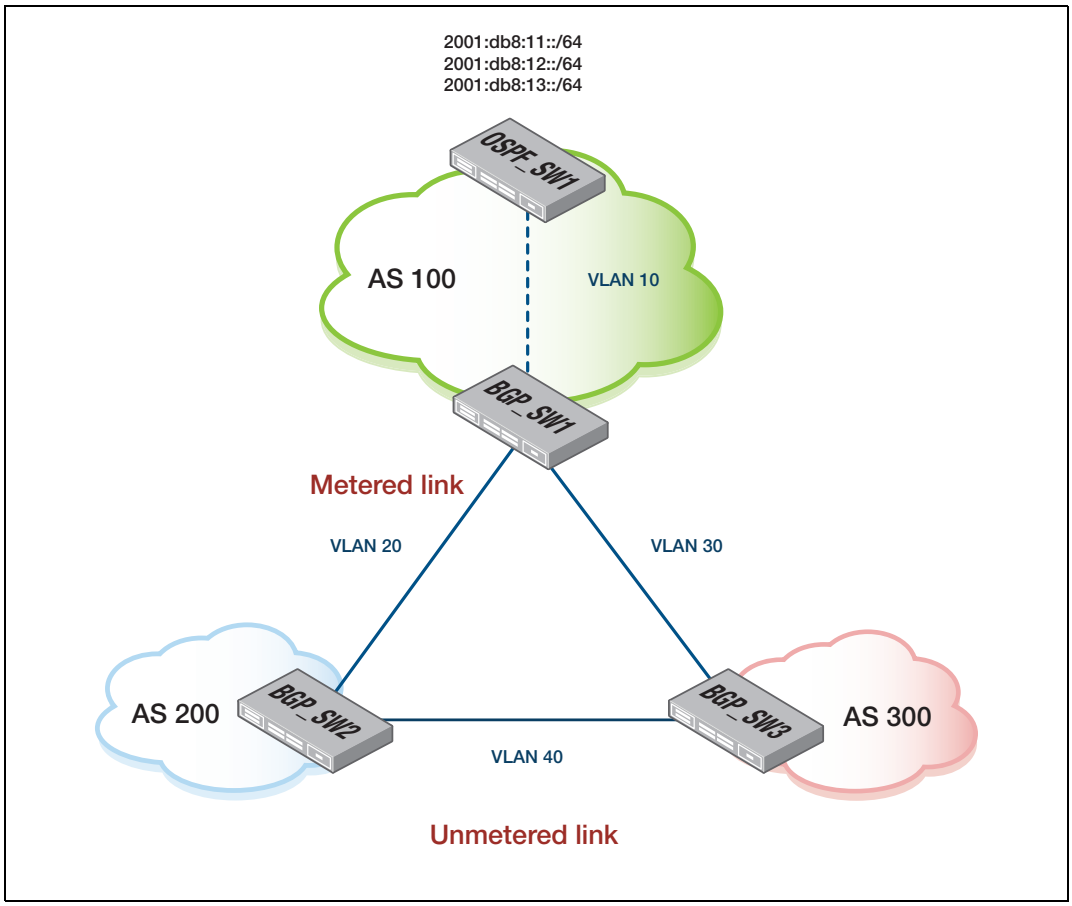
Configuration of BGP_SW3

Note: Route filtering has been applied to BGP so the static routes to the neighbor's loopback is not advertised.

```
BGP_SW3#

!
hostname BGP_SW3
!
vlan database
 vlan 30,40 state enable
!
interface port1.0.1
 switchport access vlan 30
!
interface port1.0.2
 switchport access vlan 40
!
interface lo
 ipv6 address 2001:db8:3::1/64
!
interface vlan30
 ipv6 address 2001:db8:30::2/64
!
interface vlan40
 ipv6 address 2001:db8:40::2/64
 ipv6 router rip
!
router ipv6 rip
 redistribute static
 distribute-list prefix FILTER_OUT out
!
router bgp 300
 bgp router-id 3.3.3.3
 no bgp default ipv4-unicast
 neighbor 2001:db8:1::1 remote-as 100
 neighbor 2001:db8:1::1 ebgp-multihop 2
 neighbor 2001:db8:1::1 update-source lo
 !
 address-family ipv6
 network 2001:db8:21::/64 backdoor
 network 2001:db8:22::/64 backdoor
 network 2001:db8:23::/64 backdoor
 redistribute static
 neighbor 2001:db8:1::1 activate
 neighbor 2001:db8:1::1 prefix-list FILTER_OUT out
 exit-address-family
!
ipv6 route 2001:db8:1::1/128 fe80::200:cdff:fe2a:eaf1 vlan30
ipv6 route 2001:db8:31::/64 Null
ipv6 route 2001:db8:32::/64 Null
ipv6 route 2001:db8:33::/64 Null
!
ipv6 prefix-list FILTER_OUT seq 10 deny 2001:db8:1::1/128
ipv6 prefix-list FILTER_OUT seq 20 permit any
!
end
```

Using weight to change path preference for all prefix(es) received into a router

Scenario



- The link between BGP_SW2 and BGP_SW1 is metered, and will incur additional costs.
- The link between BGP_SW2 and BGP_SW3 is not metered.
- The current path from AS200 to prefixes residing in AS 100 is via the metered link.
- The desired path from AS200 to prefixes in AS 100 is via the unmetered link, transiting AS 300.

Objective

The objective is for the path from AS200 via AS 300 to be selected in the BGP table as the best path.

View of BGP_SW2's BGP table prior to weight adjustment

```
BGP_SW2#show bgp ipv6
BGP table version is 3, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network             Next Hop            Metric LocPrf Weight Path
*> 2001:db8:11::/64 2001:db8:1::1           20             0 100 ?
                    2001:db8:3::1            0             0 300 100 ?
*> 2001:db8:12::/64 2001:db8:1::1           20             0 100 ?
                    2001:db8:3::1            0             0 300 100 ?
*> 2001:db8:13::/64 2001:db8:1::1           20             0 100 ?
                    2001:db8:3::1            0             0 300 100 ?

Number of prefixes 3
```

1. Set the **weight** attribute for all prefixes advertised by a BGP neighbor.

With the weight attribute, the heavier weight is preferred. All routes advertised by neighbor 2001:db8:3::1 (BGP_SW3 in AS 300) will have a weight value of 200 applied.

```
BGP_SW2(config)#router bgp 200
BGP_SW2(config-router)#address-family ipv6
BGP_SW2(config-router-af)#neighbor 2001:db8:3::1 weight 200
```

2. Clear the BGP neighbor session to apply the change.

```
BGP_SW2#clear bgp ipv6 2001:db8:3::1
```

Verify by viewing the:

- BGP neighbor output for neighbor 2001:db8:3::1 (weight value is specified).
- BGP table (preferred path and weight value is specified).
- Routing table.

```
BGP_SW2#show bgp ipv6 neighbors 2001:db8:3::1
BGP neighbor is 2001:db8:3::1, remote AS 300, local AS 200, external link
  BGP version 4, remote router ID 3.3.3.3
  BGP state = Established, up for 00:02:32
  Last read 00:02:32, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    4-Octet ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 66 messages, 1 notifications, 0 in queue
  Sent 67 messages, 3 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
  Update source is lo
 For address family: IPv6 Unicast
  BGP table version 11, neighbor version 11
  Index 2, Offset 0, Mask 0x4
  Community attribute sent to this neighbor (both)
 Weight 200
  3 accepted prefixes
  0 announced prefixes

 Connections established 5; dropped 4
  External BGP neighbor may be up to 2 hops away.
Local host: 2001:db8:2::1, Local port: 51342
Foreign host: 2001:db8:3::1, Foreign port: 179
Nexthop: 2.2.2.2
Nexthop global: 2001:db8:2::1
Nexthop local: ::
BGP connection: non shared network
Last Reset: 00:02:33, due to BGP Notification sent
Notification Error Message: (Cease/Other Configuration Change.
```

```
BGP_SW2#show bgp ipv6
BGP table version is 9, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:11::/64 2001:db8:3::1            0           200 300 100 ?
*                   2001:db8:1::1           20             0 100 ?
*> 2001:db8:12::/64 2001:db8:3::1            0           200 300 100 ?
*                   2001:db8:1::1           20             0 100 ?
*> 2001:db8:13::/64 2001:db8:3::1            0           200 300 100 ?
*                   2001:db8:1::1           20             0 100 ?

Number of prefixes 3
```

```
BGP_SW2#show ipv6 route database
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
       IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
       > - selected route, * - FIB route, p - stale info
Timers: Uptime

S  *> 2001:db8:1::1/128 [1/0] via fe80::200:cdff:fe2a:eaf1, vlan20, 00:26:00
C  *> 2001:db8:2::/64 via ::, lo, 00:26:00
S  *> 2001:db8:3::1/128 [1/0] via fe80::200:cdff:fe2a:e635, vlan40, 00:26:00
B  *> 2001:db8:11::/64 [20/0] via 2001:db8:3::1 (recursive via fe80::200:cdff:fe2a:e635), vlan40,
00:01:36
B  *> 2001:db8:12::/64 [20/0] via 2001:db8:3::1 (recursive via fe80::200:cdff:fe2a:e635), vlan40,
00:01:36
B  *> 2001:db8:13::/64 [20/0] via 2001:db8:3::1 (recursive via fe80::200:cdff:fe2a:e635), vlan40,
00:01:36
C  *> 2001:db8:20::/64 via ::, vlan20, 00:25:53
C  *> 2001:db8:40::/64 via ::, vlan40, 00:25:53
C  *  fe80::/64 via ::, vlan40, 00:25:53
C  *  fe80::/64 via ::, vlan20, 00:25:53
C  *> fe80::/64 via ::, lo, 00:26:00
```

This chapter covers the following topics:

- Load sharing to a single ISP from a single dual-homed router using the max-paths command
- Change path preference for selected prefix(es) received into a router using the weight command to achieve load balancing
- Using BGP multi hop to achieve load sharing
- Using BGP communities to adjust Local Preference

CHAPTER 7

Load Sharing Examples

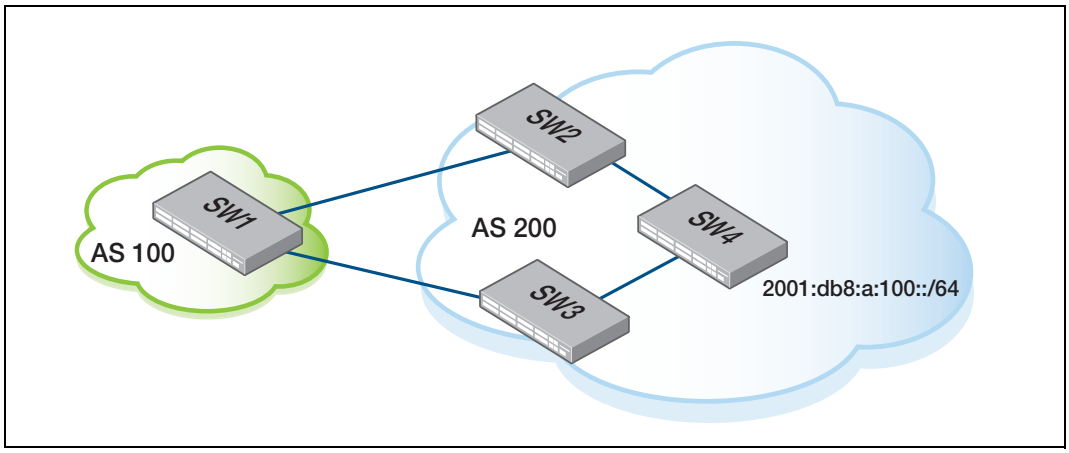
By default BGP always selects the best path to reach a destination. On occasions when there is more than one exit point or path from an AS, it may be preferred to instead perform non-default load sharing of traffic across the redundant paths whilst both paths are available, and for all traffic to exit the AS via the single available path when other redundant paths are down.

The following section describes how to configure some load sharing scenarios.

Load sharing to a single ISP from a single dual-homed router using the max-paths command

Scenario

- SW1 is a customer connected to an ISP (AS 200) with two links for redundancy and load sharing.



Default behaviour

- By default BGP will select only one path from the available equal cost paths to install in the Routing Information Base (RIB).
- In solutions where equal cost paths are desired, BGP must select more than one best path to be installed in the RIB.
- The max-paths command overrides the default BGP behaviour, allowing BGP to install up to the maximum defined equal cost paths in the RIB.

Implementation example

- When load sharing across two or more equal cost paths is desired.

Objective

- Configure BGP to install two equal cost paths in the RIB to achieve load sharing.

View of SW1's route table before max-paths has been configured for BGP. Currently BGP is only installing one path (the best path) in the RIB by default.

```
SW1#show ipv6 route database
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
       IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
       > - selected route, * - FIB route, p - stale info
Timers: Uptime
C *> 2001:db8:1::/64 via ::, vlan100, 2d16h08m
C *> 2001:db8:10::/64 via ::, vlan10, 2d16h08m
C *> 2001:db8:20::/64 via ::, vlan20, 2d16h08m
B *> 2001:db8:100::/64 [20/0] via fe80::eecd:6dff:fe20:c23e, vlan20, 2d16h07m
C * fe80::/64 via ::, vlan20, 2d16h08m
C * fe80::/64 via ::, vlan10, 2d16h08m
C * fe80::/64 via ::, vlan100, 2d16h08m
C *> fe80::/64 via ::, lo, 2d16h08m
```

1. Configure the max-paths command under the BGP IPv6 Address Family

- Up to 64 equal cost paths are supported by AlliedWare Plus.

```
SW1(config)#router bgp 100
SW1(config-router)#address-family ipv6
SW1(config-router-af)#max-paths ebgp 2
```

Note: Defining the maximum equal cost paths for iBGP can also be achieved by using:

```
SW1(config-router-af)#max-paths ibgp ?
<2-64> Supported multipath numbers
```

Note: To remove the configuration, enter:

```
SW1(config-router-af)#no max-paths ebgp
```

Verify by viewing the:

- Routing table on SW1, (two equal cost paths should be installed).
- Route information for 2001:db8:100::/64, (two next hop addresses should be present for the route).
- BGP route entry. (Note both paths show multipath-candidate, and installed.)

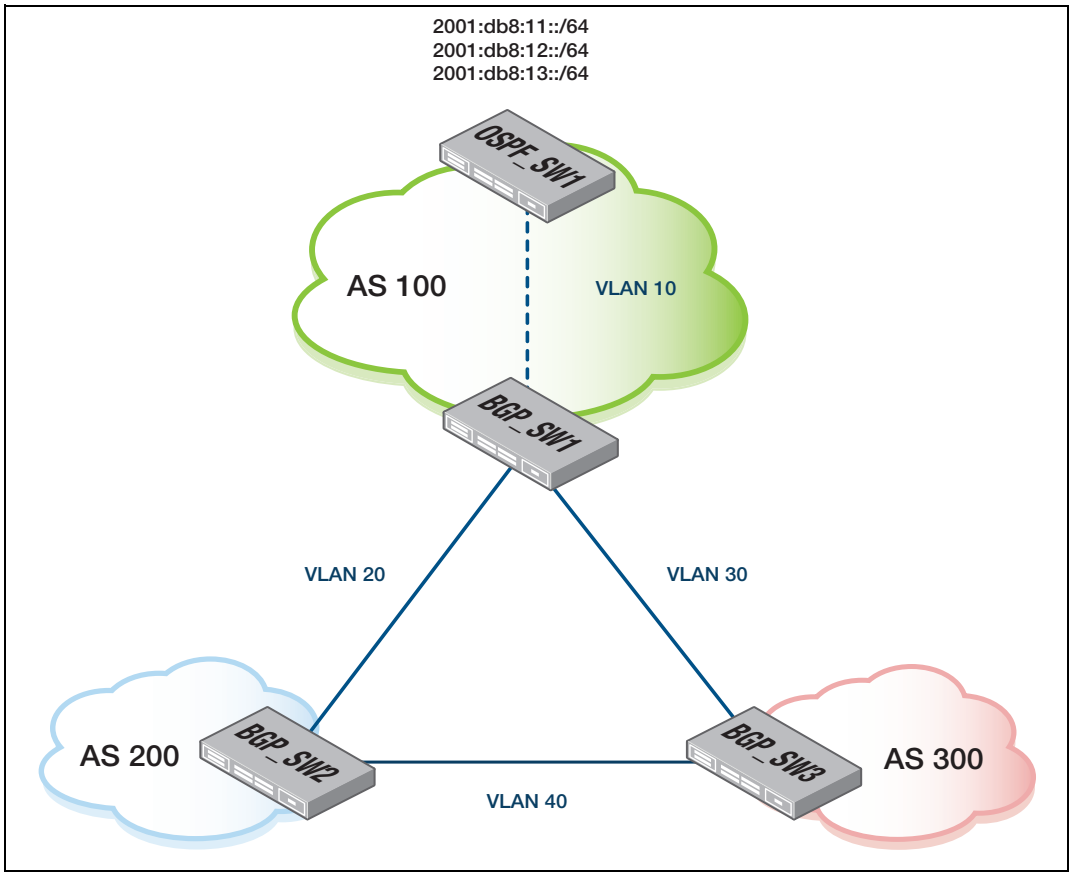
```
SW1#show ipv6 route 2001:db8:100::/64
Routing entry for 2001:db8:100::/64
Known via "bgp", distance 20, metric 0, best
Last update 00:03:45 ago
* via 2001:db8:10::2, vlan10
* via 2001:db8:20::2, vlan20
```

```
SW1#show bgp ipv6 2001:db8:100::/64
BGP routing table entry for 2001:db8:100::/64
Paths: (2 available, best #2, table Default-IP-Routing-Table)
 Advertised to non peer-group peers:
 2001:db8:10::2
 200
   2001:db8:10::2(fe80::eecd:6dff:fe20:c26b) from 2001:db8:10::2 (2.2.2.2)
  (fe80::eecd:6dff:fe20:c26b)
   Origin IGP metric 0, localpref 100,valid,external, multipath-candidate, installed
   Last update: Fri Jun 21 04:48:57 2013
200
  2001:db8:20::2(fe80::eecd:6dff:fe20:c23e) from 2001:db8:20::2 (3.3.3.3)
 (fe80::eecd:6dff:fe20:c23e)
  Origin IGP metric 0, localpref 100, valid, external, multipath-candidate, installed, best
  Last update: Fri Jun 21 04:48:49 2013
```

Using the weight command to achieve load balancing

By changing path preference for selected prefix(es) received into a router

Scenario



- The link between BGP_SW2 and BGP_SW1 is heavily utilized with all traffic from AS 200 to prefixes 2001:db8:11: :/64, 2001:db8:12: :/64, and 2001:db8:13: :/64 in AS 100 using this link by default.

Objective

- Load sharing is required to more evenly balance the egress traffic from AS 200 towards AS100.
- The desired path to prefix 2001:db8:11::/64 and 2001:db8:13::/64 is the link to AS 100.
- The desired path to prefix 2001:db8:12::/64 is via the alternate link via AS 300.

View of BGP_SW2's BGP table prior to weight adjustment

```
BGP_SW2#show bgp ipv6
BGP table version is 3, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network              Next Hop            Metric LocPrf Weight Path
*> 2001:db8:11::/64  2001:db8:1::1           20             0 100 ?
                     2001:db8:3::1            0             0 300 100 ?
*> 2001:db8:12::/64  2001:db8:1::1           20             0 100 ?
                     2001:db8:3::1            0             0 300 100 ?
*> 2001:db8:13::/64  2001:db8:1::1           20             0 100 ?
                     2001:db8:3::1            0             0 300 100 ?

Number of prefixes 3
```

1. Create an Access List or a Prefix List to classify prefix 2001:db8:12::/64.

Option 1:

```
BGP_SW2(config)#ipv6 access-list standard REMOTE_NETWORK_LIST permit
   2001:db8:12::/64
```

Option 2:

```
BGP_SW2(config)#ipv6 prefix-list REMOTE_NETWORK_LIST seq 10 permit
   2001:db8:12::/64
```

2. Create a Route Map to match on the ACL or Prefix List and apply a weight value for the matching prefix(es).

- Sequence 10 will apply weight 200 to prefixes matching the ACL / Prefix List.
- Sequence 20 will permit all other prefixes, leaving the default value unchanged.

Option 1: (ACL used)

```
BGP_SW2(config)#route-map REMOTE_NETWORK permit 10
BGP_SW2(config-route-map)#match ipv6 address REMOTE_NETWORK_LIST
BGP_SW2(config-route-map)#set weight 200
BGP_SW2(config-route-map)#route-map REMOTE_NETWORK permit 20
```

Option 2: (Prefix list used)

```
BGP_SW2(config)#route-map REMOTE_NETWORK permit 10
BGP_SW2(config-route-map)#match ipv6 address prefix-list
   REMOTE_NETWORK_LIST
BGP_SW2(config-route-map)#set weight 200
BGP_SW2(config-route-map)#route-map REMOTE_NETWORK permit 20
```

3. Apply the Route Map for prefixes advertised by BGP_SW3 in AS 300.

```
BGP_SW2(config)#router bgp 200
BGP_SW2(config-router)#address-family ipv6
BGP_SW2(config-router-af)#neighbor 2001:db8:3::1 route-map
   REMOTE_NETWORK in
```

4. Clear BGP session to neighbor 2001:db8:3::1 (BGP_SW3 in AS 300).

```
BGP_SW2#clear bgp ipv6 2001:db8:3::1
```

Verify by viewing the:

- BGP neighbor output for neighbor 2001:db8:3::1 (weight value is specified).
- BGP table (preferred path and weight value is specified).
- Routing table.

```
BGP_SW2#show bgp ipv6 neighbors 2001:db8:3::1
BGP neighbor is 2001:db8:3::1, remote AS 300, local AS 200, external link
  BGP version 4, remote router ID 3.3.3.3
  BGP state = Established, up for 00:01:52
  Last read 00:01:52, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    4-Octet ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 136 messages, 1 notifications, 0 in queue
  Sent 138 messages, 5 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
  Update source is lo
 For address family: IPv6 Unicast
  BGP table version 27, neighbor version 27
  Index 2, Offset 0, Mask 0x4
  Community attribute sent to this neighbor (both)
  Inbound path policy configured
  Route map for incoming advertisements is *REMOTE_NETWORK
  3 accepted prefixes
  2 announced prefixes

 Connections established 7; dropped 6
  External BGP neighbor may be up to 2 hops away.
Local host: 2001:db8:2::1, Local port: 33369
Foreign host: 2001:db8:3::1, Foreign port: 179
Nexthop: 2.2.2.2
Nexthop global: 2001:db8:2::1
Nexthop local: ::
BGP connection: non shared network
Last Reset: 00:01:53, due to BGP Notification sent
Notification Error Message: (Cease/Other Configuration Change.)
```

```
BGP_SW2#show bgp ipv6
BGP table version is 3, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network             Next Hop            Metric LocPrf Weight Path
*> 2001:db8:11::/64 2001:db8:1::1           20             0 100 ?
                    2001:db8:3::1            0             0 300 100 ?
*> 2001:db8:12::/64 2001:db8:1::1           20             0 100 ?
                    2001:db8:3::1            0           200 300 100 ?
*> 2001:db8:13::/64 2001:db8:1::1           20             0 100 ?
                    2001:db8:3::1            0             0 300 100 ?

Number of prefixes 3
```

```
BGP_SW2#show ipv6 route database
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
       IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
       > - selected route, * - FIB route, p - stale info
Timers: Uptime

S  *> 2001:db8:1::1/128 [1/0] via fe80::200:cdff:fe2a:eaf1, vlan20, 00:54:44
C  *> 2001:db8:2::/64 via ::, lo, 00:54:44
S  *> 2001:db8:3::1/128 [1/0] via fe80::200:cdff:fe2a:e635, vlan40, 00:54:44
B  *> 2001:db8:11::/64 [20/20] via 2001:db8:1::1 (recursive via fe80::200:cdff:fe2a:eaf1),
vlan20, 00:22:07
B  *> 2001:db8:12::/64 [20/0] via 2001:db8:3::1 (recursive via fe80::200:cdff:fe2a:e635), vlan40,
00:02:27
B  *> 2001:db8:13::/64 [20/20] via 2001:db8:1::1 (recursive via fe80::200:cdff:fe2a:eaf1),
vlan20, 00:22:07
C  *> 2001:db8:20::/64 via ::, vlan20, 00:54:37
C  *> 2001:db8:40::/64 via ::, vlan40, 00:54:37
C  *  fe80::/64 via ::, vlan40, 00:54:37
C  *  fe80::/64 via ::, vlan20, 00:54:37
C  *> fe80::/64 via ::, lo, 00:54:44
```

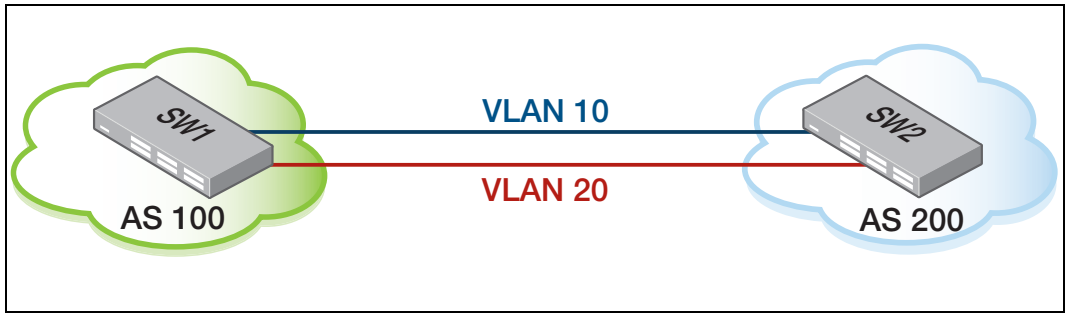
BGP_SW2 configuration

```
BGP_SW2#

!
hostname BGP_SW2
!
vlan database
 vlan 20,40 state enable
!
interface port1.0.1
 switchport access vlan 20
!
interface port1.0.2
 switchport access vlan 40
!
interface lo
 ipv6 address 2001:db8:2::1/64
!
interface vlan20
 ipv6 address 2001:db8:20::2/64
!
interface vlan40
 ipv6 address 2001:db8:40::1/64
!
router bgp 200
 bgp router-id 2.2.2.2
 no bgp default ipv4-unicast
 neighbor 2001:db8:1::1 remote-as 100
 neighbor 2001:db8:1::1 ebgp-multihop 2
 neighbor 2001:db8:1::1 update-source lo
 neighbor 2001:db8:3::1 remote-as 300
 neighbor 2001:db8:3::1 ebgp-multihop 2
 neighbor 2001:db8:3::1 update-source lo
 !
 address-family ipv6
 neighbor 2001:db8:1::1 activate
 neighbor 2001:db8:3::1 activate
 neighbor 2001:db8:3::1 route-map REMOTE_NETWORK in
 exit-address-family
!
ipv6 route 2001:db8:1::1/128 fe80::200:cdff:fe2a:eaf1 vlan20
ipv6 route 2001:db8:3::1/128 fe80::200:cdff:fe2a:e635 vlan40
!
ipv6 prefix-list REMOTE_NETWORK_LIST seq 10 permit 2001:db8:12::/64
!
route-map REMOTE_NETWORK permit 10
 match ipv6 address prefix-list REMOTE_NETWORK_LIST
 set weight 200
!
route-map REMOTE_NETWORK permit 20
!
end
```

Using BGP multi hop to achieve load sharing

Load sharing is accomplished via recursive lookup.

Scenario



Use

Achieve load sharing when there are equal cost links available to BGP.

Implementation example

Configure each BGP peer to use the internal loopback interface as the update source and neighbor address. Each device is connected via two separate links, providing physical link redundancy.

The loopback interface configured within each device is virtual, without any association to an external interface. The loopback interface is always up and is always reachable, and can be configured with one or more IPv4 or IPv6 addresses.

In addition to load sharing, peering to the remote loopback prevents unnecessary flapping between the BGP peers, since there is more than one physical path to the remote peer loopback interface.

Objective

Setup basic BGP load sharing by using the loopback for the BGP peering, and equal cost paths between the eBGP peers.

1. Setup VLANs on each switch.

```
SW1(config)#vlan database
SW1(config-vlan)#vlan 10
SW1(config-vlan)#vlan 20

SW2(config)#vlan database
SW2(config-vlan)#vlan 10
SW2(config-vlan)#vlan 20
```

2. Setup IPv6 interfaces.

```
SW1(config)#interface vlan10
SW1(config-if)#ipv6 address 2001:db8:10::1/64
SW1(config-if)#interface vlan20
SW1(config-if)#ipv6 address 2001:db8:20::1/64

SW2(config)#interface vlan10
SW2(config-if)#ipv6 address 2001:db8:10::2/64
SW2(config-if)#interface vlan20
SW2(config-if)#ipv6 address 2001:db8:20::2/64
```

3. Assign the switch ports to the VLANs.

```
SW1(config)#interface port1.0.11
SW1(config-if)#switchport access vlan 10
SW1(config-if)#interface port1.0.12
SW1(config-if)#switchport access vlan 20

SW2(config)#interface port1.0.11
SW2(config-if)#switchport access vlan 10
SW2(config-if)#interface port1.0.12
SW2(config-if)#switchport access vlan 20
```

4. Setup IPv6 loopback interfaces to use for the BGP peering.

```
SW1(config)#interface lo
SW1(config-if)#ipv6 address 2001:db8:1::1/128
SW2(config)#interface lo
SW2(config-if)#ipv6 address 2001:db8:2::1/128
```

5. Setup static IPv6 routes to reach the neighbor's loopback interface.

Note: It is recommended to use the IPv6 link-local address of the neighbor as the Next Hop address.

```
SW1(config)#ipv6 route 2001:db8:2::1/128 fe80::eecd:6dff:fe20:c244
   vlan10
SW1(config)#ipv6 route 2001:db8:2::1/128 fe80::eecd:6dff:fe20:c244
   vlan20
SW2(config)#ipv6 route 2001:db8:1::1/128 fe80::eecd:6dff:fe48:e55b
   vlan10
SW2(config)#ipv6 route 2001:db8:1::1/128 fe80::eecd:6dff:fe48:e55b
   vlan20
```

6. Configure BGP to peer using loopback interfaces.

```
SW1(config)#router bgp 100
SW1(config-router)#bgp router-id 1.1.1.1
SW1(config-router)#no bgp default ipv4-unicast
SW1(config-router)#neighbor 2001:db8:2::1 remote-as 200
SW1(config-router)#neighbor 2001:db8:2::1 update-source lo
SW1(config-router)#neighbor 2001:db8:2::1 ebgp-multihop 2
SW1(config-router)#address-family ipv6
SW1(config-router-af)#neighbor 2001:db8:2::1 activate

SW2(config)#router bgp 200
SW2(config-router)#bgp router-id 2.2.2.2
SW2(config-router)#no bgp default ipv4-unicast
SW2(config-router)#neighbor 2001:db8:1::1 remote-as 100
SW2(config-router)#neighbor 2001:db8:1::1 update-source lo
SW2(config-router)#neighbor 2001:db8:1::1 ebgp-multihop 2
SW2(config-router)#address-family ipv6
SW2(config-router-af)#neighbor 2001:db8:1::1 activate
```

7. Advertise prefixes to AS 200.

```
SW1(config)#router bgp 100
SW1(config-router)#address-family ipv6
SW1(config-router-af)#network 2001:db8:100::/64
SW1(config-router-af)#network 2001:db8:200::/64
SW1(config-router-af)#network 2001:db8:300::/64
```

Verify

- Using a recursive route lookup, AlliedWare Plus can load share traffic flows over the two equal cost links.
- View this using the IPv6 routing table.

```
SW2#show ipv6 route database
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
> - selected route, * - FIB route, p - stale info
Timers: Uptime
S *> 2001:db8:1::1/128 [1/0] via fe80::eecd:6dff:fe48:e55b, vlan10, 00:09:37
*> [1/0] via fe80::eecd:6dff:fe48:e55b, vlan20, 00:09:37
C *> 2001:db8:2::1/128 via ::, lo, 00:19:44
C *> 2001:db8:10::/64 via ::, vlan10, 00:08:08
C *> 2001:db8:20::/64 via ::, vlan20, 00:07:59
B *> 2001:db8:100::/64 [20/0] via 2001:db8:1::1 (recursive via
fe80::eecd:6dff:fe48:e55b, vlan10 via fe80::eecd:6dff:fe48:e55b, vlan20), 00:00:10
B *> 2001:db8:200::/64 [20/0] via 2001:db8:1::1 (recursive via
fe80::eecd:6dff:fe48:e55b, vlan10 via fe80::eecd:6dff:fe48:e55b, vlan20), 00:00:10
B *> 2001:db8:300::/64 [20/0] via 2001:db8:1::1 (recursive via
fe80::eecd:6dff:fe48:e55b, vlan10 via fe80::eecd:6dff:fe48:e55b, vlan20), 00:00:10
C * fe80::/64 via ::, vlan20, 00:07:59
C * fe80::/64 via ::, vlan10, 00:08:08
C *> fe80::/64 via ::, lo, 00:20:00
```

How it works

- Notice the BGP table below only has one Next Hop, (`2001:db8:1::1`) for each network.
- This is because as far as BGP is concerned, the destination networks are reached via the single Next Hop address, which is the BGP peer's loopback address. So, from BGP perspective there is only a single path to the peer.
- When AlliedWare Plus does a route lookup for the Next Hop address, it knows of two equal cost paths to reach the neighbor loopback address, therefore installing both paths in the FIB. The result is all BGP prefixes learned from the peer are reachable using both recursive paths to the Next Hop address.

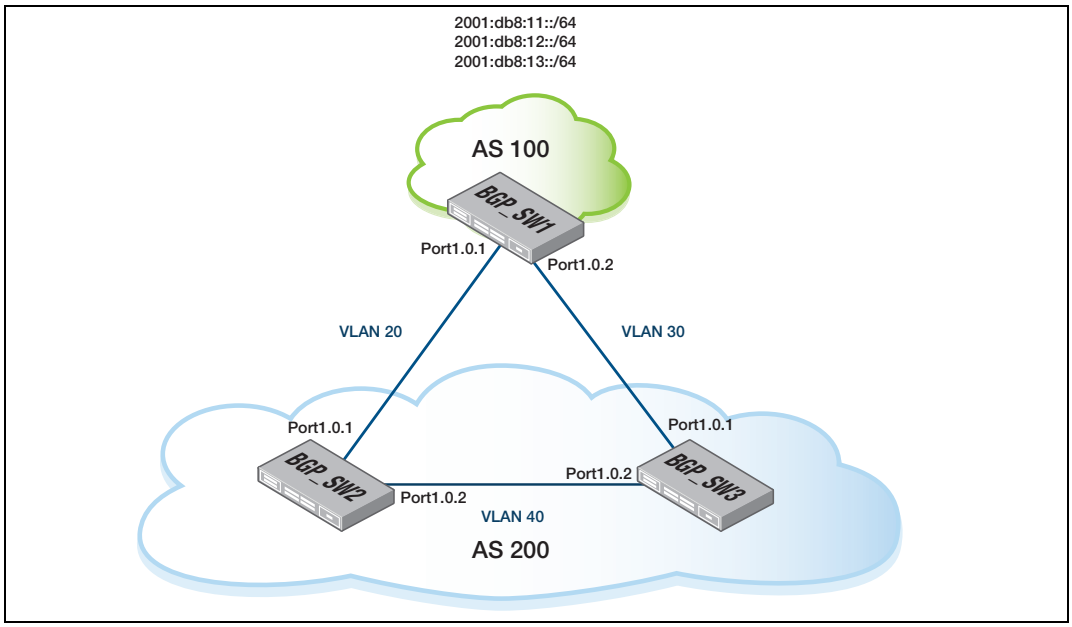
BGP table view

```
SW2#show bgp ipv6
BGP table version is 5, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network             Next Hop            Metric LocPrf Weight Path
*> 2001:db8:100::/64
                    2001:db8:1::1            0             0 100 i
*> 2001:db8:200::/64
                    2001:db8:1::1            0             0 100 i
*> 2001:db8:300::/64
                    2001:db8:1::1            0             0 100 i
Number of prefixes 3
```

Using BGP communities to adjust Local Preference

Scenario

The use of BGP communities to control routing policies is defined in RFC 1998.



- BGP_SW1 is located in AS 100.
- BGP_SW2 and BGP_SW3 is located in AS 200.
- AS 100 is the customer.
- AS 200 is the service provider.

Objectives

- The service provider is able to utilize both links to AS 100 by using the BGP community attribute in conjunction with Local Preference.
- The customer is to announce prefixes with a community attribute, which will inform the service provider which link to use to send data to AS 100.
- Inbound traffic for customer prefixes 2001:db8:11::/64 and 2001:db8:13::/64 are to use the link between BGP_SW1 and BGP_SW3
- Inbound traffic for customer prefixes 2001:db8:12::/64 is to use the link between BGP_SW1 and BGP_SW2.

Service provider policy

- If a prefix is announced with a community attribute of 100:150, then the service provider sets the local preference for those routes as 150.
- If a prefix is announced with a community attribute of 100:100, then the service provider sets the local preference for those routes as 100.

View of BGP_SW2's and BGP_SW3's BGP table prior to route filtering

Currently the best path to AS 100 networks is via each switches directly connected link.

```
BGP_SW2#show bgp ipv6
BGP table version is 91, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
* i2001:db8:11::/64 2001:db8:1::1            0    100      0 100 i
*>                  2001:db8:1::1            0             0 100 i
* i2001:db8:12::/64 2001:db8:1::1            0    100      0 100 i
*>                  2001:db8:1::1            0             0 100 i
* i2001:db8:13::/64 2001:db8:1::1            0    100      0 100 i
*>                  2001:db8:1::1            0             0 100 i

Number of prefixes 3
```

```
BGP_SW3#show bgp ipv6
BGP table version is 12, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:11::/64 2001:db8:1::1            0             0 100 i
* i                  2001:db8:1::1            0    100      0 100 i
*> 2001:db8:12::/64 2001:db8:1::1            0             0 100 i
* i                  2001:db8:1::1            0    100      0 100 i
*> 2001:db8:13::/64 2001:db8:1::1            0             0 100 i
* i                  2001:db8:1::1            0    100      0 100 i

Number of prefixes 3
```

1. Create an ACL/Prefix List to classify networks to be matched in route maps on customer BGP_SW1 (AS 100).

- Two ACLs or Prefix Lists must be created.
- Each list will be used to match on in the route maps, so the community attribute can be set.
- For this example only ACLs will be used.

```
BGP_SW1(config)#ipv6 access-list standard BGP_SW2_PATH_LIST permit
   2001:db8:12::/64
BGP_SW1(config)#ipv6 access-list standard BGP_SW3_PATH_LIST permit
   2001:db8:11::/64
BGP_SW1(config)#ipv6 access-list standard BGP_SW3_PATH_LIST permit
   2001:db8:13::/64
```

2. Create two route maps to match prefixes in the ACLs or Prefix Lists to set a community value.

- Sequence 10 will match prefixes in the preferred path list for the link and set the community attribute value 100:150.
- Sequence 20 will match prefixes in the unpreferred path list for the link and set the community attribute value 100:100.

```
BGP_SW1(config)#route-map BGP_SW2_PATH permit 10
BGP_SW1(config-route-map)#match ipv6 address BGP_SW2_PATH_LIST
BGP_SW1(config-route-map)#set community 100:150

BGP_SW1(config-route-map)#route-map BGP_SW2_PATH permit 20
BGP_SW1(config-route-map)#match ipv6 address BGP_SW3_PATH_LIST
BGP_SW1(config-route-map)#set community 100:100

BGP_SW1(config)#route-map BGP_SW3_PATH permit 10
BGP_SW1(config-route-map)#match ipv6 address BGP_SW3_PATH_LIST
BGP_SW1(config-route-map)#set community 100:150

BGP_SW1(config-route-map)#route-map BGP_SW3_PATH permit 20
BGP_SW1(config-route-map)#match ipv6 address BGP_SW2_PATH_LIST
BGP_SW1(config-route-map)#set community 100:100
```

3. Apply the route map when peering with BGP_SW2 and BGP_SW3.

```
BGP_SW1(config)#router bgp 100
BGP_SW1(config-router)#address-family ipv6
BGP_SW1(config-router-af)#neighbor 2001:db8:2::1 send-community
BGP_SW1(config-router-af)#neighbor 2001:db8:2::1 route-map BGP_SW2_PATH out
BGP_SW1(config-router-af)#neighbor 2001:db8:3::1 send-community
BGP_SW1(config-router-af)#neighbor 2001:db8:3::1 route-map BGP_SW3_PATH out
```

4. Setup community lists on BGP_SW2 and BGP_SW3.

```
BGP_SW2(config)#ip community-list 1 permit 100:150
BGP_SW2(config)#ip community-list 2 permit 100:100
BGP_SW3(config)#ip community-list 1 permit 100:150
BGP_SW3(config)#ip community-list 2 permit 100:100
```

5. Setup route maps to match on the community lists and set a local preference on BGP_SW2 and BGP_SW3.

- Sequence 10 matches community list 1, and on match will set a local preference of 150.
- Sequence 20 matches community list 2, and on match will set a local preference of 100.
- Sequence 30 will permit all other updates that do not match either list.

```
BGP_SW2(config)#route-map BGP_SW2_PATH permit 10
BGP_SW2(config-route-map)#match community 1
BGP_SW2(config-route-map)#set local-preference 150
BGP_SW2(config-route-map)#route-map BGP_SW2_PATH permit 20
BGP_SW2(config-route-map)#match community 2
BGP_SW2(config-route-map)#set local-preference 100
BGP_SW2(config-route-map)#route-map BGP_SW2_PATH permit 30

BGP_SW3(config)#route-map BGP_SW3_PATH permit 10
BGP_SW3(config-route-map)#match community 1
BGP_SW3(config-route-map)#set local-preference 150

BGP_SW3(config-route-map)#route-map BGP_SW3_PATH permit 20
BGP_SW3(config-route-map)#match community 2
BGP_SW3(config-route-map)#set local-preference 100
BGP_SW3(config-route-map)#route-map BGP_SW3_PATH permit 30
```

6. Setup apply the route map when peering with BGP_SW1 and set the **next-hop-self** parameter for iBGP peers.

```
BGP_SW2(config)#router bgp 200
BGP_SW2(config-router)#address-family ipv6
BGP_SW2(config-router-af)#neighbor 2001:db8:3::1 next-hop-self
BGP_SW2(config-router-af)#neighbor 2001:db8:1::1 route-map BGP_SW2_PATH in

BGP_SW3(config)#router bgp 200
BGP_SW3(config-router)#address-family ipv6
BGP_SW3(config-router-af)#neighbor 2001:db8:2::1 next-hop-self
BGP_SW3(config-router-af)#neighbor 2001:db8:1::1 route-map BGP_SW3_PATH in
```

Verify by:

- Viewing the BGP table on BGP_SW2 and BGP_SW3.
  - Best path to prefix 2001:db8:12::/64 in AS 100 is via BGP_SW2
  - Best path to prefix 2001:db8:11::/64 and 2001:db8:13::/64 in AS 100 is via BGP_SW3
- Viewing BGP route maps.
- Viewing BGP neighbor output.

```
BGP_SW2#show bgp ipv6
BGP table version is 5, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*>i2001:db8:11::/64 2001:db8:3::1            0    150      0 100 i
*                   2001:db8:1::1            0    100      0 100 i
*> 2001:db8:12::/64 2001:db8:1::1            0    150      0 100 i
*>i2001:db8:13::/64 2001:db8:3::1            0    150      0 100 i
*                   2001:db8:1::1            0    100      0 100 i
BGP_SW3#show bgp ipv6
BGP table version is 18, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:11::/64 2001:db8:1::1            0    150      0 100 i
*>i2001:db8:12::/64 2001:db8:2::1            0    150      0 100 i
*                   2001:db8:1::1            0    100      0 100 i
*> 2001:db8:13::/64 2001:db8:1::1            0    150      0 100 i

Number of prefixes 3
BGP_SW2#show bgp ipv6 route-map BGP_SW2_PATH
BGP table version is 8, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*>i2001:db8:11::/64 2001:db8:3::1            0    150      0 100 i
*                   2001:db8:1::1            0    100      0 100 i
*> 2001:db8:12::/64 2001:db8:1::1            0    150      0 100 i
*>i2001:db8:13::/64 2001:db8:3::1            0    150      0 100 i
*                   2001:db8:1::1            0    100      0 100 i

Number of prefixes 3
BGP_SW3#show bgp ipv6 route-map BGP_SW3_PATH
BGP table version is 20, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:11::/64 2001:db8:1::1            0    150      0 100 i
*>i2001:db8:12::/64 2001:db8:2::1            0    150      0 100 i
*                   2001:db8:1::1            0    100      0 100 i
*> 2001:db8:13::/64 2001:db8:1::1            0    150      0 100 i

Number of prefixes 3
```

BGP_SW1 configuration

```
BGP_SW1#
!
hostname BGP_SW1
!
ipv6 access-list standard BGP_SW2_PATH_LIST permit 2001:db8:12::/64
ipv6 access-list standard BGP_SW3_PATH_LIST permit 2001:db8:11::/64
ipv6 access-list standard BGP_SW3_PATH_LIST permit 2001:db8:13::/64
!
vlan database
 vlan 10,20,30 state enable
!
interface port1.0.1
 switchport access vlan 20
!
interface port1.0.2
 switchport access vlan 30
!
interface port1.0.24
 switchport access vlan 10
!
interface lo
 ipv6 address 2001:db8:1::1/64
!
interface vlan10
 ipv6 address 2001:db8:10::1/64
!
interface vlan20
 ipv6 address 2001:db8:20::1/64
!
interface vlan30
 ipv6 address 2001:db8:30::1/64
 ipv6 enable
!
router bgp 100
 bgp router-id 1.1.1.1
 no bgp default ipv4-unicast
 neighbor 2001:db8:2::1 remote-as 200
 neighbor 2001:db8:2::1 ebgp-multihop 2
 neighbor 2001:db8:2::1 update-source lo
 neighbor 2001:db8:3::1 remote-as 200
 neighbor 2001:db8:10::2 remote-as 10
 !
 address-family ipv6
 network 2001:db8:11::/64
 network 2001:db8:12::/64
 network 2001:db8:13::/64
 neighbor 2001:db8:2::1 activate
 neighbor 2001:db8:2::1 route-map BGP_SW2_PATH out
 neighbor 2001:db8:3::1 activate
 neighbor 2001:db8:3::1 route-map BGP_SW3_PATH out
 neighbor 2001:db8:10::2 activate
 exit-address-family
!
ipv6 route 2001:db8:2::1/128 fe80::209:41ff:fefb:c323 vlan20
ipv6 route 2001:db8:3::1/128 fe80::200:cdff:fe2a:e635 vlan30
!
route-map BGP_SW2_PATH permit 10
 match ipv6 address BGP_SW2_PATH_LIST
 set community 100:150
!
route-map BGP_SW2_PATH permit 20
 match ipv6 address BGP_SW3_PATH_LIST
 set community 100:100
!
route-map BGP_SW3_PATH permit 10
 match ipv6 address BGP_SW3_PATH_LIST
 set community 100:150
!
route-map BGP_SW3_PATH permit 20
 match ipv6 address BGP_SW2_PATH_LIST
 set community 100:100
!
end
```

BGP_SW2 configuration

```
BGP_SW2#
!
hostname BGP_SW2
!
vlan database
 vlan 20,40 state enable
!
interface port1.0.1
 switchport access vlan 20
!
interface port1.0.2
 switchport access vlan 40
!
interface lo
 ipv6 address 2001:db8:2::1/64
!
interface vlan20
 ipv6 address 2001:db8:20::2/64
!
interface vlan40
 ipv6 address 2001:db8:40::1/64
!
router bgp 200
 bgp router-id 2.2.2.2
 neighbor 2001:db8:1::1 remote-as 100
 neighbor 2001:db8:1::1 ebgp-multihop 2
 neighbor 2001:db8:1::1 update-source lo
 neighbor 2001:db8:3::1 remote-as 200
 neighbor 2001:db8:3::1 update-source lo
 !
 address-family ipv6
 neighbor 2001:db8:1::1 activate
 neighbor 2001:db8:1::1 route-map BGP_SW2_PATH in
 neighbor 2001:db8:3::1 activate
 neighbor 2001:db8:3::1 next-hop-self
 exit-address-family
!
ip community-list 1 permit 100:150
ip community-list 2 permit 100:100
!
ipv6 route 2001:db8:1::1/128 fe80::200:cdff:fe2a:eaf1 vlan20
ipv6 route 2001:db8:3::1/128 fe80::200:cdff:fe2a:e635 vlan40
!
route-map BGP_SW2_PATH permit 10
 match community 1
 set local-preference 150
!
route-map BGP_SW2_PATH permit 20
 match community 2
 set local-preference 100
!
route-map BGP_SW2_PATH permit 30
!
end
```

BGP_SW3 configuration

```
BGP_SW3#

!
hostname BGP_SW3
!
vlan database
 vlan 30,40 state enable
!
interface port1.0.1
 switchport access vlan 30
!
interface port1.0.2
 switchport access vlan 40
!
interface lo
 ipv6 address 2001:db8:3::1/64
!
interface vlan30
 ipv6 address 2001:db8:30::2/64
!
interface vlan40
 ipv6 address 2001:db8:40::2/64
!
router bgp 200
 bgp router-id 3.3.3.3
 no bgp default ipv4-unicast
 neighbor 2001:db8:1::1 remote-as 100
 neighbor 2001:db8:1::1 ebgp-multihop 2
 neighbor 2001:db8:1::1 update-source lo
 neighbor 2001:db8:2::1 remote-as 200
 neighbor 2001:db8:2::1 update-source lo
 !
 address-family ipv6
 neighbor 2001:db8:1::1 activate
 neighbor 2001:db8:1::1 route-map BGP_SW3_PATH in
 neighbor 2001:db8:2::1 activate
 neighbor 2001:db8:2::1 next-hop-self
 exit-address-family
!
ip community-list 1 permit 100:150
ip community-list 2 permit 100:100
!
ipv6 route 2001:db8:1::1/128 fe80::200:cdff:fe2a:eaf1 vlan30
ipv6 route 2001:db8:2::1/128 fe80::209:41ff:fefb:c323 vlan40
!
route-map BGP_SW3_PATH permit 10
 match community 1
 set local-preference 150
!
route-map BGP_SW3_PATH permit 20
 match community 2
 set local-preference 100
!
route-map BGP_SW3_PATH permit 30
!
end
```


BGP4+

This chapter covers the following topics:

- Adjusting the minimum interval between BGP Update messages within an AS
- Limiting the number of permitted prefixes advertised by a neighbor
- Setup dampening for a prefix using a route map
- Using a route map to apply different dampening policies to separate prefixes
- Using BGP synchronization to avoid reachability issues in a transit AS
- Permit routes that originate from a remote AS with the same ASN

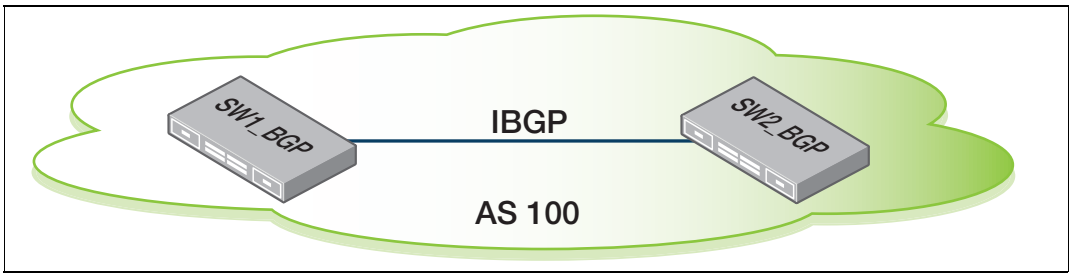
CHAPTER 8

Session Management Options

This section describes a number of other useful BGP4+ processes and command options for managing BGP4+ peering sessions.

Adjusting the minimum interval between BGP Update messages within an AS

- Command: as-origination-interval



Use

The as-origination-interval command enables the minimum interval between BGP UPDATE messages that report changes within the speakers own AS.

Difference between route advertisement interval and AS origination interval:

- The route advertisement interval determines the minimum amount of time between BGP UPDATE messages for the advertisement or withdrawal of routes learned from BGP peers.
- The as origination interval determines the minimum amount of time between BGP UPDATE messages sent from the BGP speaker for advertisement or withdrawal of routes originated by the speaker

Note: This interval only specifies the "minimum" time in seconds.

More information

http://tools.ietf.org/html/rfc4271 (section 9.2.1.2)

Objective

- Change the minimum interval between BGP UPDATE messages that report changes within the speaker (A1's) own AS.
- BGP UPDATE messages to iBGP peers should be sent at least 30 seconds apart or more.
- The default as-origination timer is 15 seconds in AlliedWare Plus (as recommended in RFC 4271).

1. Change the default AS origination.

```
BGP_SW1(config)#router bgp 100
BGP_SW1(config-router)#neighbor 2001:db8:a:10::2 as-origination-
   interval 30
```

Optional: Revert to default value of 15 seconds.

Simply negate the command to reset the timer to 15 seconds.

```
BGP_SW1(config-router)#no neighbor 2001:db8:a:10::2 as-origination-interval
```

Verify by:

- Viewing the BGP debug.
- Look for the [FSM] AS-Origination Timer Expiry messages

Notice

It takes 30 seconds for the AS-Origination Timer to expire for an UPDATE to be sent advertising the originated prefix to be withdrawn.

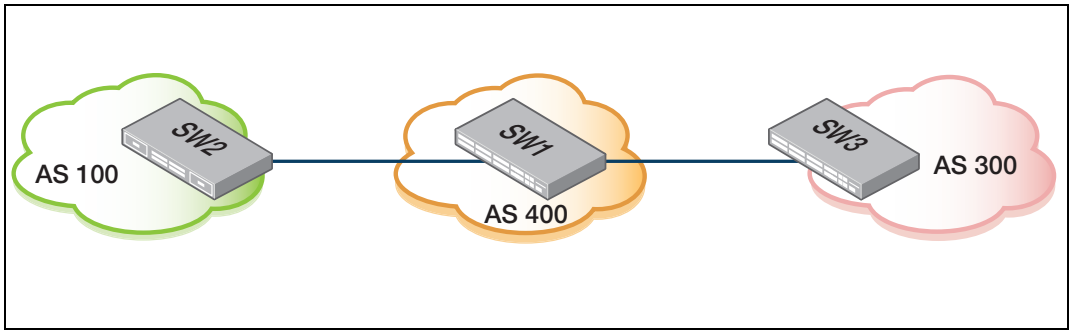
In the example, we are originating a route on SW1_BGP. The AS Origination timer must expire before this change internal to the AS can be advertised to iBGP peers.

```
14:32:59 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [ENCODE] Msg-Hdr: Type 2
14:32:59 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [ENCODE] Update MP Reach:
Prefix 2001:db8:a:90::/64
14:32:59 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [ENCODE] Update: AFI/SAFI (2/
1) Tot-attr-len 64
14:32:59 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [ENCODE] Update: Msg #28 Size
87
A1(config-router-af)#no network 2001:db8:a:90::/64
14:33:00 A1 IMISH[3558]: no network 2001:db8:a:90::/6
14:33:00 A1 BGP[1970]: [SFL] route count decreased to 24
14:33:03 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [FSM] Routeadv Timer Expiry)
14:33:03 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [FSM] State: Established Event:
34
14:33:04 A1 BGP[1970]: [RIB] Scanning BGP RIB...
14:33:04 A1 BGP[1970]: NSM Message Header
14:33:04 A1 BGP[1970]: VR ID: 0
14:33:04 A1 BGP[1970]: VRF ID: 0
14:33:04 A1 BGP[1970]: Message type: IPv6 Route (32)
14:33:04 A1 BGP[1970]: Message length: 76
14:33:04 A1 BGP[1970]: Message ID: 0x00000076
14:33:04 A1 BGP[1970]: NSM IPv6 route add
14:33:04 A1 BGP[1970]: Flags: 1
14:33:04 A1 BGP[1970]: Route: 2001:db8:a:70::/64
14:33:04 A1 BGP[1970]: Type: 7
14:33:04 A1 BGP[1970]: Metric: 3
14:33:04 A1 BGP[1970]: Distance: 110
14:33:04 A1 BGP[1970]: Nexthop: fe80::209:41ff:fefb:c323 ifindex 310
14:33:04 A1 BGP[1970]: [NSM] Verified NH 2001:db8:a:70::1 with NSM
14:33:04 A1 BGP[1970]: [RIB] Scanning BGP Network Routes...
```

[Continued on next page...]

```
14:33:07 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [FSM] Routeadv Timer Expiry)
14:33:07 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [FSM] State: Established Event: 34
14:33:11 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [FSM] Routeadv Timer Expiry)
14:33:11 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [FSM] State: Established Event: 34
14:33:15 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [FSM] Routeadv Timer Expiry)
14:33:15 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [FSM] State: Established Event: 34
14:33:18 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [FSM] Routeadv Timer Expiry)
14:33:18 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [FSM] State: Established Event: 34
14:33:18 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [DECODE] Msg-Hdr: type 4, length 19
14:33:18 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [DECODE] Msg-Hdr: Requesting immediate Read (0)
14:33:18 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [DECODE] KAlive: Received!
14:33:18 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [FSM] State: Established Event: 26
14:33:19 A1 BGP[1970]: [RIB] Scanning BGP Network Routes...
14:33:21 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [FSM] Routeadv Timer Expiry)
14:33:21 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [FSM] State: Established Event: 34
14:33:22 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [FSM] Keep-alive-Timer Expiry
14:33:22 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [FSM] State: Established Event: 11
14:33:22 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [ENCODE] Msg-Hdr: Type 4
14:33:22 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [ENCODE] Keepalive: 136 KAlive msg(s) sent
14:33:25 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [FSM] Routeadv Timer Expiry)
14:33:25 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [FSM] State: Established Event: 34
14:33:28 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [FSM] AS-Origination Timer Expiry
14:33:28 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [FSM] State: Established Event: 33
14:33:29 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [FSM] Routeadv Timer Expiry)
14:33:29 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [FSM] State: Established Event: 34
14:33:29 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [ENCODE] Msg-Hdr: Type 2
14:33:29 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [ENCODE] Update Withdrawn: Prefix
2001:db8:a:90::/64
14:33:29 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [ENCODE] Update: AFI/SAFI (2/1) Tot-attr-len 16
14:33:29 A1 BGP[1970]: 2001:db8:a:10::2-Outgoing [ENCODE] Update: Msg #29 Size 39
```

Limiting the number of permitted prefixes advertised by a neighbor



- Command: maximum-prefix

Objective

- Configure a maximum number of prefixes allowed to be learned by device SW3 from neighbor SW1 in AS 400.

Use

- Mainly used for eBGP peering, but can be used for iBGP peering too.
- Useful when there is limited memory available on the device, and the number of prefixes learned from a neighbor needs to be limited.

Note:

- The maximum-prefix command will check the number of prefixes advertised from a neighbor.
- If the maximum-prefix value is exceeded, the router will terminate the neighbor session.
- If this happens all prefixes learned will be cleared, and the state will remain Idle until the advertised prefix value drops to a permitted value.
- If the session is taken down because the maximum-prefix limit has been exceeded, you must reset the session using the clear bgp ipv6 command.

Alternative to session being shutdown when maximum-prefix limit is reached

- A warning only message can be sent. The session will not be shutdown.
- Threshold value (percentage). When the number of prefixes reaches the threshold, (percentage of prefix limit), a warning is generated.

Example 1

```
SW3(config-router)#neighbor 2001:db8::1 maximum-prefix 23 ?
  <1-100>       threshold-value, 1 to 100 percent
  warning-only  Only give warning message when limit is exceeded
  <cr>
```

Example 2

```
neighbor 2001:db8::1 maximum-prefix 50 50
(When 50% of the max number of prefixes have been learned (50% of 50 – 25) generate
a warning.
```

View of SW3's BGP table prior to maximum-prefix configuration

```
SW3#show bgp ipv6
BGP table version is 4, local router ID is 3.3.3.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> ::/0             2001:db8:d::1            0             0 400 100 i
*> 2001:db8:5::/64  2001:db8:d::1            0             0 400 ?
*> 2001:db8:6::/64  2001:db8:d::1            0             0 400 ?
*>i2001:db8:c:20::/64
                    2001:db8:c:10::2(fe80::eecd:6dff:fe20:c23e)
                                             0    100      0 i
*> 2001:db8:d::1/128
                    2001:db8:d::1            0             0 400 ?
*> 2001:db8:d:10::/64
                    2001:db8:d::1            0             0 400 ?
*> 2001:db8:d:20::/64
                    2001:db8:d::1            0             0 400 ?
*> 2001:db8:d:30::/64
                    2001:db8:d::1            0             0 400 i
*> 2001:db8:d:40::/64
                    2001:db8:d::1            0             0 400 i

Number of prefixes 9
```

1. Set a maximum-prefix value of 7 for the session to neighbor SW1.

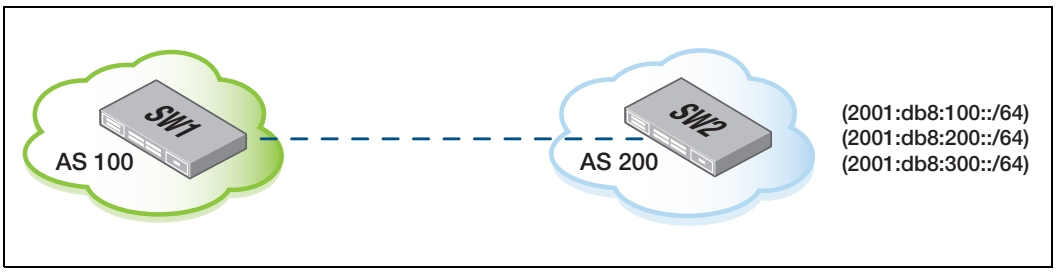
```
SW3(config)#router bgp 300
SW3(config-router)#address-family ipv6
SW3(config-router-af)#neighbor 2001:db8:d::1 maximum-prefix 7
```

Verify by viewing the:

- BGP table on SW3. (Notice that because the maximum prefixes has been exceeded, the prefixes have been cleared.)
- State of the session to SW1 using the command: show bgp ipv6 summary.

```
SW3#show bgp ipv6
BGP table version is 8, local router ID is 3.3.3.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*>i2001:db8:c:20::/64
                    2001:db8:c:10::2(fe80::eecd:6dff:fe20:c23e)
                                             0    100      0 i
Number of prefixes 1

SW3#show bgp ipv6 summary
BGP router identifier 3.3.3.1, local AS number 300
BGP table version is 8
1 BGP AS-PATH entries
0 BGP community entries

Neighbor           V     AS   MsgRc  MsgSnt TblVer InOutQ Up/Down State/PfxRcd
2001:db8:b::1      4    200     103    166        0 0/0    never    Active
2001:db8:c:10::2   4    300     351    384        8 0/0    00:03:53        1
2001:db8:d::1      4    400     320    314        0 0/0    never    Idle-PfxCt
```

Setup dampening for a prefix using a route map



Use

Use a route map to match specific prefixes to setup BGP route dampening.

Objective

Using a route map, match and set BGP route dampening for the prefix 2001:db8:100::/64.

1. Setup an IPv6 ACL to permit the 2001:db8:100::/64 prefix.

```
SW1(config)#ipv6 access-list standard DAMPENING_LIST permit
   2001:db8:100::/64
SW1(config)#ipv6 access-list standard DAMPENING_LIST deny any
```

2. Setup a route map to match on the IPv6 ACL and set BGP dampening.

- Sequence 10 will match on the IPv6 ACL and set BGP dampening for the learned prefix.

```
SW1(config)#route-map DAMPENING_MAP permit 10
SW1(config-route-map)#match ipv6 address DAMPENING_LIST
SW1(config-route-map)#set dampening
```

3. Setup BGP dampening and apply the route map to set dampening only for the specific prefix.

```
SW1(config)#router bgp 100
SW1(config-router)#address-family ipv6
SW1(config-router-af)#bgp dampening route-map DAMPENING_MAP
```

Verify

- Verify on SW1 by viewing the successful match for the route map.
- Only the 2001:db8:100::/64 prefix will have dampening applied.

```
SW1#show bgp ipv6 route-map DAMPENING_MAP
BGP table version is 3, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network          Next Hop            Metric LocPrf Weight Path
*> 2001:db8:100::/64
                 2001:db8:10::2(fe80::eecd:6dff:fe20:c073)
                                          0             0 200 i
Number of prefixes 1
```

Route flap simulation

Simulate route flapping to see the flap statistics and dampening.

Simulation #1

- Route flap counter increments.
- Penalty increments with each flap.
- Route is dampened when penalty is greater than the suppress limit.

```
SW1#show bgp ipv6 2001:db8:100::/64
10:57:20 SW1 IMISH[16209]: show bgp ipv6 2001:db8:100::/64
BGP routing table entry for 2001:db8:100::/64
Paths: (1 available, no best path)
Not advertised to any peer
200, (history entry)
2001:db8:10::2(fe80::eecd:6dff:fe20:c073) from 2001:db8:10::2 (2.2.2.2)
(fe80::eecd:6dff:fe20:c073)
  Origin IGP metric 0, localpref 100, external
  Dampinfo: penalty 1000, flapped 1 times in 00:00:03
  Last update: Thu Jul 11 10:52:11 2013
```

```
SW1#show bgp ipv6 dampening flap-statistics
10:59:15 SW1 IMISH[16209]: show bgp ipv6 dampening flap-statistics
BGP table version is 18, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network          From                Flaps    Duration  Reuse  Path
*> 2001:db8:100::/64
                 2001:db8:10::2(fe80::eecd:6dff:fe20:c073)
                                       1   00:01:58          200 i
Number of prefixes 1
```

```
SW1#show bgp ipv6
BGP table version is 19, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network             Next Hop              Metric LocPrf Weight Path
h 2001:db8:100::/64
                    2001:db8:10::2(fe80::eecd:6dff:fe20:c073)
                                                0             0 200 i
*> 2001:db8:200::/64
                    2001:db8:10::2(fe80::eecd:6dff:fe20:c073)
                                                0             0 200 i
*> 2001:db8:300::/64
                    2001:db8:10::2(fe80::eecd:6dff:fe20:c073)
                                                0             0 200 i
Number of prefixes 3
```

```
SW1#show bgp ipv6 dampening dampened-paths
11:01:12 SW1 IMISH[16209]: show bgp ipv6 dampening dampened-paths
BGP table version is 20, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network              From                  Reuse     Path
*d 2001:db8:100::/64
                    2001:db8:10::2(fe80::eecd:6dff:fe20:c073)
                                          00:27:00  200 i
Number of prefixes 1
```

```
SW1#show bgp ipv6 dampening flap-statistics
11:01:25 SW1 IMISH[16209]: show bgp ipv6 dampening flap-statistics
BGP table version is 21, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network              From                    Flaps  Duration  Reuse    Path
*d 2001:db8:100::/64
                     2001:db8:10::2(fe80::eecd:6dff:fe20:c073)
                                               3    00:04:08 00:26:50   200 i
Number of prefixes 1
```

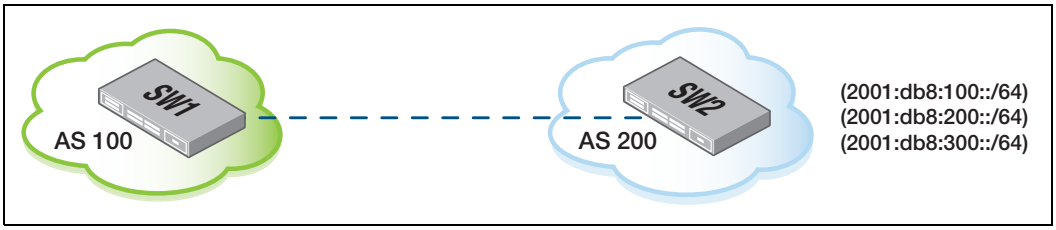
Simulation #2

If we flap BOTH the 2001:db8:100::/64 and 2001:db8:200::/64 prefixes, dampening is only applied to the 2001:db8:100::/64 prefix.

- Notice the 2001:db8:200::/64 prefix is simply removed from the BGP table.
- Notice the 2001:db8:100::/64 prefix has incurred penalty and has history.

```
SW1#show bgp ipv6
BGP table version is 5, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop                Metric LocPrf Weight Path
 h 2001:db8:100::/64
                   2001:db8:10::2(fe80::eecd:6dff:fe20:c073)
                                                0             0  200 i
*> 2001:db8:300::/64
                   2001:db8:10::2(fe80::eecd:6dff:fe20:c073)
                                                0             0  200 i
Number of prefixes 2
```

Using a route map to apply different dampening policies to separate prefixes



Use

Use a route map to match specific prefixes to apply different dampening policies.

Objective

- Using a route map, match and set policy #1 dampening values for the prefixes 2001:db8:100::/64 and 2001:db8:300::/64.
- Using a route map, match and set policy #2 dampening values for the prefix 2001:db8:200::/64.

Policy #1:

- half-life = 10
- reuse = 700
- suppress = 2000
- max suppress time = 60

Policy #2:

- half-life = 15
- reuse = 1000
- suppress = 5000
- max suppress time = 60

1. Setup IPv6 ACLs to list the permitted prefixes.

```
SW1(config)#ipv6 access-list standard DAMP_LIST#1 permit
   2001:db8:100::/64
SW1(config)#ipv6 access-list standard DAMP_LIST#1 permit
   2001:db8:300::/64
SW1(config)#ipv6 access-list standard DAMP_LIST#1 deny any
SW1(config)#ipv6 access-list standard DAMP_LIST#2 permit
   2001:db8:200::/64
SW1(config)#ipv6 access-list standard DAMP_LIST#1 deny any
```

2. Setup a route map to match on the IPv6 ACL and set BGP dampening.

- Sequence 10 will match on IPv6 ACL DAMP_LIST#1 and apply the dampening values for policy #1.
- Sequence 20 will match on IPv6 ACL DAMP_LIST#1 and apply the dampening values for policy #2.

```
SW1(config)#route-map DAMPENING_MAP permit 10
SW1(config-route-map)#match ipv6 address DAMP_LIST#1
SW1(config-route-map)#set dampening 10 700 2000 60
SW1(config-route-map)#
SW1(config-route-map)#route-map DAMPENING_MAP permit 20
SW1(config-route-map)#match ipv6 address DAMP_LIST#2
SW1(config-route-map)#set dampening 10 1000 5000 60
```

3. Setup BGP dampening and apply the route map to set dampening only for the specific prefix.

```
SW1(config)#router bgp 100
SW1(config-router)#address-family ipv6
SW1(config-router-af)#bgp dampening route-map DAMPENING_MAP
```

Verify

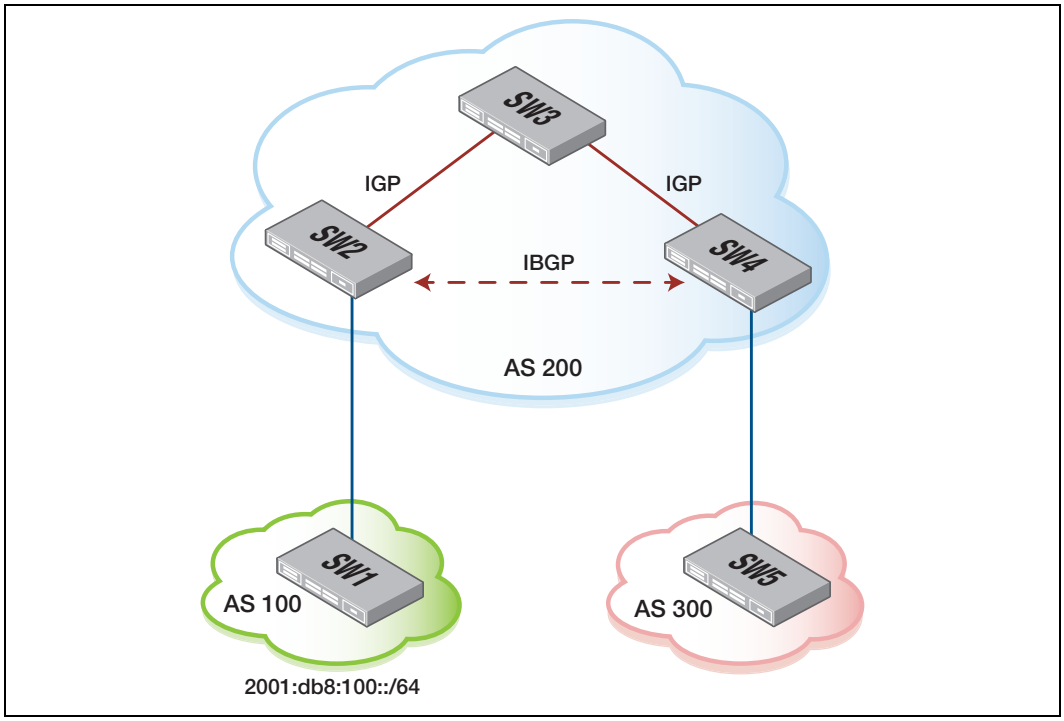
View the dampening parameters; there should be two different policies present.

```
SW1#show bgp ipv6 dampening parameters
 dampening route-map DAMPENING_MAP
 Dampening Control Block(s)       :
  Reachability Half-Life time     : 10 min
  Reuse penalty                   : 700
  Suppress penalty                : 2000
  Max suppress time               : 60 min
  Un-reachability Half-Life time  : 10 min
  Max penalty (ceil)              : 44800
  Min penalty (floor)             : 350

  Reachability Half-Life time     : 10 min
  Reuse penalty                   : 1000
  Suppress penalty                : 5000
  Max suppress time               : 60 min
  Un-reachability Half-Life time  : 10 min
  Max penalty (ceil)              : 64000
  Min penalty (floor)             : 500
```

Using BGP synchronization to avoid reachability issues in a transit AS

Scenario

- iBGP peering exists between SW2 and SW4.
- RIPng is used as the IGP in transit AS 200.



Use

- Prevent reachability issues by ensuring BGP does not advertise a route to an external peer, until the route is learned via an IGP.
- BGP synchronization is disabled by default in AlliedWare Plus.

Implementation example

- A transit AS consists of network devices between BGP speakers, which are not running BGP and therefore use an IGP (such as OSPFv3 or RIPng) to propagate routes through the transit network.

BGP synchronization rule

- A BGP speaker will not advertise a prefix learned via an iBGP peer to external peers, until the prefix is learned locally or via an IGP first.

Objective

- Setup BGP synchronization to ensure SW4 learns the 2001:db8:100::/64 prefix via an IGP first, before it advertises the same prefix learned via the iBGP peer (SW2), to the external peer (SW5).

BGP synchronization problem – BGP synchronization is disabled

A false sense of reachability is created as SW5 has learned the prefix from SW4, but the destination network is unreachable as SW3 has no route to the destination.

Part 1

- Connectivity fails due to BGP synchronization problem.
- SW1 has advertised the 2001:db8:100::/64 prefix to SW2 in AS 200 via the eBGP peering.
- SW2 has advertised this to SW4 via the iBGP peering.
- SW4 advertised this prefix to SW5 in AS 300 via the eBGP peering.
- SW5 attempts to reach the address 2001:db8:100::1 unsuccessfully.

```
SW5#show bgp ipv6
BGP table version is 2, local router ID is 4.4.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network          Next Hop                    Metric LocPrf Weight Path
*> 2001:db8:100::/64
                 2001:db8:40::1(fe80::eecd:6dff:fe48:e55b)
                                                  0             0 200 100 i

Number of prefixes 1

SW5#ping ipv6 2001:db8:100::1
PING 2001:db8:100::1(2001:db8:100::1) 56 data bytes
--- 2001:db8:100::1 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 3999ms
```

Part 2

- Connectivity fails because SW4 sends the traffic destined for 2001:db8:100::1 to the next hop address which resides at SW3, but SW3 does not have a route to the 2001:db8:100::/64 network learned via an IGP or locally.

```
SW3#show ipv6 route database
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
       IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
       > - selected route, * - FIB route, p - stale info
Timers: Uptime
R *> 2001:db8:10::/64 [120/2] via fe80::eecd:6dff:fe20:c073, vlan20, 00:02:27
R    2001:db8:20::/64 [0/1] via ::, vlan20, 00:02:54
C *> 2001:db8:20::/64 via ::, vlan20, 00:02:54
R    2001:db8:30::/64 [0/1] via ::, vlan30, 00:03:21
C *> 2001:db8:30::/64 via ::, vlan30, 00:03:21
R *> 2001:db8:40::/64 [120/2] via fe80::eecd:6dff:fe48:e55b, vlan30, 00:02:55
C *> fe80::/64 via ::, vlan30, 00:03:21
C *  fe80::/64 via ::, vlan20, 00:03:28
```

Part 3

- Because BGP synchronization is disabled, SW4 still installs the 2001:db8:100::/64 prefix and advertises this to SW5.
- This creates the false sense of reachability.

```
SW4#show ipv6 route database
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
> - selected route, * - FIB route, p - stale info
Timers: Uptime
R *> 2001:db8:10::/64 [120/3] via fe80::eecd:6dff:fe48:e54e, vlan30, 00:02:36
R *> 2001:db8:20::/64 [120/2] via fe80::eecd:6dff:fe48:e54e, vlan30, 00:03:04
R 2001:db8:30::/64 [0/1] via ::, vlan30, 00:03:32
C *> 2001:db8:30::/64 via ::, vlan30, 00:03:33
R 2001:db8:40::/64 [0/1] via ::, vlan40, 00:03:32
C *> 2001:db8:40::/64 via ::, vlan40, 00:03:32
B *> 2001:db8:100::/64 [200/0] via 2001:db8:20::1 (recursive via
fe80::eecd:6dff:fe48:e54e, vlan30), 00:01:40
C *> fe80::/64 via ::, vlan30, 00:03:33
C * fe80::/64 via ::, vlan40, 00:03:39
C * fe80::/64 via ::, vlan30, 00:03:39
```

1. Enable BGP Synchronization.

```
SW4(config)#router bgp 200
SW4(config-router)#address-family ipv6
SW4(config-router-af)#synchronization
```

Verify synchronization is working

- Notice the prefix is no longer present in the SW4 routing table, as there is no local route or IGP route to the destination on SW4.
- Notice SW4 is no longer advertising the unreachable network to external peer SW5.

```
SW4#show ipv6 route database
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
> - selected route, * - FIB route, p - stale info
Timers: Uptime
R *> 2001:db8:10::/64 [120/3] via fe80::eecd:6dff:fe48:e54e, vlan30, 00:06:24
R *> 2001:db8:20::/64 [120/2] via fe80::eecd:6dff:fe48:e54e, vlan30, 00:06:52
R    2001:db8:30::/64 [0/1] via ::, vlan30, 00:07:20
C *> 2001:db8:30::/64 via ::, vlan30, 00:07:21
R    2001:db8:40::/64 [0/1] via ::, vlan40, 00:07:20
C *> 2001:db8:40::/64 via ::, vlan40, 00:07:20
C *> fe80::/64 via ::, vlan30, 00:07:21
C *  fe80::/64 via ::, vlan40, 00:07:27
C *  fe80::/64 via ::, vlan30, 00:07:27


SW5#show bgp ipv6
BGP table version is 3, local router ID is 4.4.4.4
No BGP prefix exists
```

2. Make the 2001:db8:100::/64 prefix reachable to SW4 via an IGP.

- Setup SW2 to redistribute the prefix learned from AS 100 into the IGP used in AS 200 (RIPng).

```
SW2(config)#router ipv6 rip
SW2(config-router)#redistribute bgp
```

Verify synchronization is working

- By viewing the RIB entry for the 2001:db8:100::/64 prefix via the IGP on SW4.
- By viewing the prefix on SW4 (Notice the "Synchronized" message).
- Notice SW4 has also re-advertised the 2001:db8:100://64 prefix to SW5.
- Notice the address 2001:db8:100::1 is now reachable from SW5 in AS 300.

```
SW4#show ipv6 route database
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
       IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
> - selected route, * - FIB route, p - stale info
Timers: Uptime
R *> 2001:db8:10::/64 [120/3] via fe80::eecd:6dff:fe48:e54e, vlan30, 00:08:20
R *> 2001:db8:20::/64 [120/2] via fe80::eecd:6dff:fe48:e54e, vlan30, 00:08:48
R    2001:db8:30::/64 [0/1] via ::, vlan30, 00:09:16
C *> 2001:db8:30::/64 via ::, vlan30, 00:09:17
R    2001:db8:40::/64 [0/1] via ::, vlan40, 00:09:16
C *> 2001:db8:40::/64 via ::, vlan40, 00:09:16
B    2001:db8:100::/64 [200/0] via 2001:db8:20::1, 00:00:21
R *> 2001:db8:100::/64 [120/3] via fe80::eecd:6dff:fe48:e54e, vlan30, 00:00:30
C *> fe80::/64 via ::, vlan30, 00:09:17
C *  fe80::/64 via ::, vlan40, 00:09:23
C *  fe80::/64 via ::, vlan30, 00:09:23
```

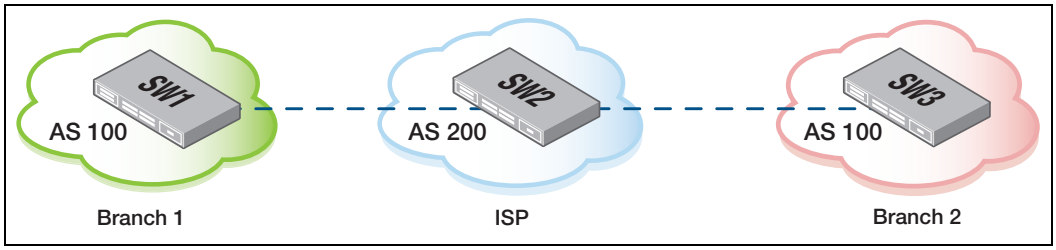
```
SW4#show bgp ipv6 2001:db8:100::/64
BGP routing table entry for 2001:db8:100::/64
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
   2001:db8:40::2
   100
    2001:db8:20::1 (metric 2) from 2001:db8:20::1 (2.2.2.2)
     Origin IGP metric 0, localpref 100, valid, internal, synchronized, best
     Last update: Wed Jul 10 02:56:18 2013
```

```
SW5#show bgp ipv6
BGP table version is 4, local router ID is 4.4.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network             Next Hop             Metric LocPrf   Weight   Path
*> 2001:db8:100::/64
                    2001:db8:40::1(fe80::eecd:6dff:fe48:e55b)
                                              0              0   200 100 i
Number of prefixes 1
```

```
SW5#ping ipv6 2001:db8:100::1
PING 2001:db8:100::1(2001:db8:100::1) 56 data bytes
64 bytes from 2001:db8:100::1: icmp_seq=1 ttl=61 time=2.04 ms
64 bytes from 2001:db8:100::1: icmp_seq=2 ttl=61 time=1.30 ms
64 bytes from 2001:db8:100::1: icmp_seq=3 ttl=61 time=1.32 ms
64 bytes from 2001:db8:100::1: icmp_seq=4 ttl=61 time=1.32 ms
64 bytes from 2001:db8:100::1: icmp_seq=5 ttl=61 time=1.33 ms

--- 2001:db8:100::1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4003ms
rtt min/avg/max/mdev = 1.303/1.467/2.049/0.294 ms
```

```
SW5#traceroute ipv6 2001:db8:100::1
traceroute6
2001:db8:100::1
traceroute to 2001:db8:100::1 (2001:db8:100::1) from 2001:db8:40::2, 30 hops
max, 16 byte packets
 1 2001:db8:40::1 (2001:db8:40::1) 1.903 ms 2.177 ms 1.901 ms
 2 2001:db8:30::1 (2001:db8:30::1) 1.846 ms 1.387 ms 0.952 ms
 3 2001:db8:20::1 (2001:db8:20::1) 1.824 ms 1.431 ms 1.923 ms
 4 2001:db8:100::1 (2001:db8:100::1) 1.856 ms 1.422 ms 0.935 ms
```

Permit routes that originate from a remote AS with the same ASN



- Command: allowas-in

Default behaviour

By default, BGP discards routes that it identifies as containing an AS loop. By default Branch 1 discards Branch 2 routes learned via ISP peer since Branch 1 and Branch 2 are assigned the same ASN 100.

"*If the AS_PATH attribute of a BGP route contains an AS loop, the BGP route should be excluded from the Phase 2 decision function. AS loop detection is done by scanning the full AS path (as specified in the AS_PATH attribute), and checking that the autonomous system number of the local system does not appear in the AS path. Operations of a BGP speaker that is configured to accept routes with its own autonomous system number in the AS path are outside the scope of this document.*" - Section 9.1.2 – RFC 4271

Use

- This command allows BGP to accept routes which contain the BGP Speaker's own AS number in the AS_PATH attribute field.
- You can specify the maximum number of times the BGP Speaker's AS number is permitted to appear in the routes AS_PATH.
- This feature simply overrides the BGP AS loop prevention mechanism.

Implementation

When branch offices use the same AS number and transit another AS (e.g. An ISP).

Objective

- Branch 1 should accept routes advertised by Branch 2.
- Because Branch 1's AS number should only appear once in received routes, the allowas-in command will be set to 1.
- For this simulation, we are assuming that Branch 2 is already correctly configured.

View of SW1's BGP neighbor session information prior to allowas-in configured:

- Notice no prefixes have been accepted.
- Debug shows: Update: Prefix 2001:db8:30::/64 denied due to as-path contains our own AS (see bottom of page 207)

```
SW1#show bgp ipv6 neighbors
BGP neighbor is 2001:db8:2::1, remote AS 200, local AS 100, external link
 BGP version 4, remote router ID 2.2.2.2
 BGP state = Established, up for 00:00:54
 Last read 00:00:54, hold time is 90, keepalive interval is 30 seconds
 Neighbor capabilities:
  Route refresh: advertised and received (old and new)
  4-Octet ASN Capability: advertised and received
  Address family IPv6 Unicast: advertised and received
 Received 4 messages, 0 notifications, 0 in queue
 Sent 4 messages, 0 notifications, 0 in queue
 Route refresh request: received 0, sent 0
 Minimum time between advertisement runs is 30 seconds
 Update source is lo
For address family: IPv6 Unicast
 BGP table version 1, neighbor version 1
 Index 1, Offset 0, Mask 0x2
 Community attribute sent to this neighbor (both)
 0 accepted prefixes
 0 announced prefixes

 Connections established 1; dropped 0
  External BGP neighbor may be up to 255 hops away.
Local host: 2001:db8:1::1, Local port: 179
Foreign host: 2001:db8:2::1, Foreign port: 56397
Nexthop: 1.1.1.1
Nexthop global: 2001:db8:1::1
Nexthop local: ::
BGP connection: non shared network
```

```
SW1#clear bgp ipv6 *
SW1#23:52:12 SW1 IMISH[3018]: clear bgp ipv6 *
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [FSM] State: Established Event:
35
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [ENCODE] Msg-Hdr: Type 3
23:52:12 SW1 BGP[1704]: % BGP-3-NOTIFICATION: sending to 2001:db8:2::1 6/6
(Cease/Other Configuration Change.) 0 data-bytes
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [FSM] State Change:
Established(6)->Idle(1)
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [FSM] Auto-Start Timer Expiry
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [FSM] State: Idle Event: 3
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [FSM] State Change: Idle(1)-
>Connect(2)
23:52:12 SW1 BGP[1704]: [NETWORK] Accept Thread: Incoming conn from host
2001:db8:2::1 (FD=16)
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [FSM] State: Connect Event: 14
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [FSM] InConnReq: Accepting...
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [NETWORK] FD=16, Sock Status: 0-
Success
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [FSM] State: Connect Event: 17
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [ENCODE] Msg-Hdr: Type 1
[continued on next page...]
```

```
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [ENCODE] Open: Ver 4 MyAS 100
Holdtime 90
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [ENCODE] Open: Msg-Size 61
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [FSM] State Change: Connect(2)-
>OpenSent(4)
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Msg-Hdr: type 1, length
61
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Msg-Hdr: Requesting
immediate Read (42)
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Open: Optional param len
32
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Open Opt: Option Type
2, Option Len 6
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Open Cap: Cap Code 1,
Cap Len 4
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Open Opt: Option Type
2, Option Len 6
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Open Cap: Cap Code 1,
Cap Len 4
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Open Opt: Option Type
2, Option Len 2
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Open Cap: Cap Code 128,
Cap Len 0
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Open Cap: RR Cap(old)
for all address-families
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Open Opt: Option Type
2, Option Len 2
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Open Cap: Cap Code 2,
Cap Len 0
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Open Cap: RR Cap(new)
for all address-families
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Open Opt: Option Type
2, Option Len 6
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Open Cap: Cap Code 65,
Cap Len 4
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [FSM] State: OpenSent Event: 19
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [ENCODE] Msg-Hdr: Type 4
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [ENCODE] Keepalive: 8 KAlive
msg(s) sent
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [FSM] State Change: OpenSent(4)-
>OpenConfirm(5)
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Msg-Hdr: type 4, length
19
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Msg-Hdr: Requesting
immediate Read (0)
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] KAlive: Received!
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [FSM] State: OpenConfirm Event:
26
23:52:12 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [FSM] State Change:
OpenConfirm(5)->Established(6)
23:52:13 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [FSM] AS-Origination Timer Expiry
23:52:13 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [FSM] Routeadv Timer Expiry)
23:52:13 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [FSM] State: Established Event:
33
23:52:13 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [FSM] State: Established Event:
34
23:52:13 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Msg-Hdr: type 2, length
74
23:52:13 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Msg-Hdr: Requesting
immediate Read (55)
23:52:13 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Update: Starting UPDATE
decoding... Bytes To Read (55), msg_size (55)
23:52:13 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [FSM] State: Established Event:
27
23:52:13 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [RIB] Update: Prefix
2001:db8:30::/64 denied due to as-path contains our own AS
```

1. Configure ‘allowas-in’ feature on SW1 in Branch 1.

- Configure under the IPv6 address family.
- Note, the default value is 3. You may also specify a specific number of times the Speaker's AS is allowed to appear in a routes AS_PATH.

```
SW1(config)#router bgp 100
SW1(config-router)#address-family ipv6
SW1(config-router-af)#neighbor 2001:db8:2::1 allowas-in
```

After "allowas-in" is configured

Update containing the prefix 2001:db8:30::/64 with the same AS number is now permitted:

```
23:58:35 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Msg-Hdr: type 2, length
74
23:58:35 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Msg-Hdr: Requesting
immediate Read (55)
23:58:35 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [DECODE] Update: Starting UPDATE
decoding... Bytes To Read (55), msg_size (55)
23:58:35 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [FSM] State: Established Event:
27
23:58:35 SW1 BGP[1704]: 2001:db8:2::1-Outgoing [RIB] Update: Received Prefix
2001:db8:30::/64
```

Verify by viewing the:

- BGP table information. Prefix 1001:db8:30::/64 (path 100 200) is now permitted into SW1 route BGP table.
- BGP neighbor information.
- BGP neighbor session summary.

```
SW1#show bgp ipv6
BGP table version is 9, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              l - labeled, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network              Next Hop             Metric   LocPrf Weight  Path
*> 2001:db8:30::/64  2001:db8:2::1             0              0  2001 00 i
Number of prefixes 1
```

```
SW1#show bgp ipv6 neighbors
 BGP neighbor is 2001:db8:2::1, remote AS 200, local AS 100, external link
  BGP version 4, remote router ID 2.2.2.2
  BGP state = Established, up for 00:21:09
 Last read 00:21:09, hold time is 90, keepalive interval is 30 seconds
 Neighbor capabilities:
   Route refresh: advertised and received (old and new)
   4-Octet ASN Capability: advertised and received
   Address family IPv6 Unicast: advertised and received
 Received 165 messages, 0 notifications, 0 in queue
 Sent 158 messages, 10 notifications, 0 in queue
 Route refresh request: received 0, sent 2
 Minimum time between advertisement runs is 30 seconds
 Update source is lo
For address family: IPv6 Unicast
 BGP table version 9, neighbor version 9
 Index 1, Offset 0, Mask 0x2
 Community attribute sent to this neighbor (both)
 1 accepted prefixes
 0 announced prefixes

 Connections established 10; dropped 9
  External BGP neighbor may be up to 255 hops away.
Local host: 2001:db8:1::1, Local port: 179
Foreign host: 2001:db8:2::1, Foreign port: 60732
Nexthop: 1.1.1.1
Nexthop: 1.1.1.1
Nexthop global: 2001:db8:1::1
Nexthop local: ::
BGP connection: non shared network
Last Reset: 00:21:09, due to BGP Notification sent
Notification Error Message: (Cease/Other Configuration Change.)
```

```
SW1#show bgp ipv6 summary
BGP router identifier 1.1.1.1, local AS number 100
BGP table version is 9
1 BGP AS-PATH entries
0 BGP community entries

Neighbor        V  AS  MsgRc MsgSnt TblVer InOutQ   Up/Down   State/PfxRcd
2001:db8:2::1   4  200   165   168       9 0/0      00:20:55         1
Number of neighbors 1
```

Optional: Specify how many times BGP Speaker's AS is permitted to appear in the AS_PATH attribute for a route.

- Value range is 1-10 (Default is 3).

```
SW1(config-router-af)#neighbor 2001:db8:2::1 allowas-in ?
<1-10> Number of occurrences of AS number
```

Optional: Revert to default value.

Simply negate the command to remove the functionality.

```
SW1(config-router-af)#no neighbor 2001:db8:2::1 allowas-in
```

This chapter covers the following topics:

- Hard resetting a BGP session
- Soft resetting a BGP neighbor session - inbound
- Soft resetting a BGP neighbor session - outbound
- Performing a "soft" update when Route Refresh is not supported on a BGP4+ peer
- Preventing termination of a BGP session when capability negotiation issues occur
- Example without "dont-capability-negotiate"
- Setup BGP graceful-restart capability for all BGP neighbors
- Using peer groups to simplify administration and conserve processing and memory

CHAPTER 9

Comparison of BGP Session Reset Methods

This section describes the differences between the BGP session reset methods.

Reset methods

- Hard Reset
- Soft Reset:
  - Soft Reconfiguration – Inbound method #1
  - Route Refresh – Inbound method #2
  - Soft Reset - Outbound

Hard reset description

- This method will completely tear down the session to the specified neighbor(s).
- The BGP session must be re-established before BGP Updates containing the advertised prefixes will be sent.
- BGP prefixes advertised by the peer(s) being reset are flushed from the BGP table.
- This method may cause traffic disruption.

Soft reset description

Inbound method #1: Soft Reconfiguration

- This method is the older soft reset option.
- Stores an unfiltered table of received prefixes from a neighbor.
- When the session to a neighbor is cleared "softly" inbound, the switch will use the stored unfiltered table to generate new updates.
- The new updates are placed in the BGP table.
- The list is generated immediately (visible in the BGP debug).
- This method may be memory intensive.
- This method should only be used when BGP neighbor does not support Route.

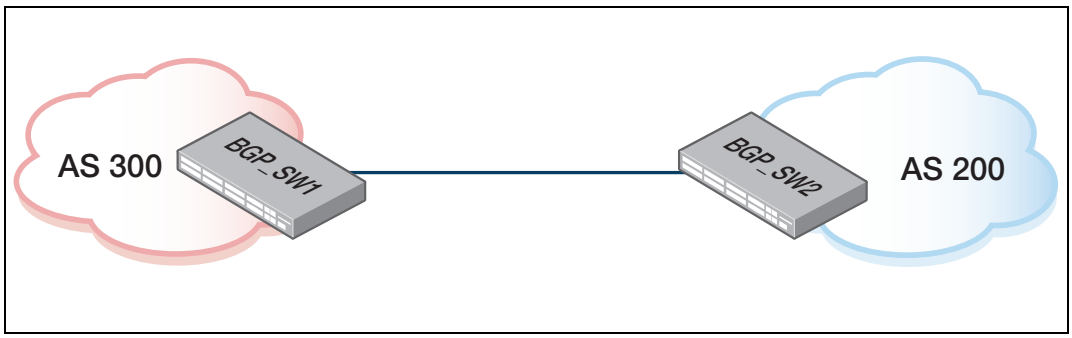
Inbound method #2: Route Refresh

- This method involves a ROUTE-REFRESH (RR) message being sent by the requesting device to the BGP peer.
- Upon receiving the RR message, BGP Updates will be sent back to the requesting device containing a new list of prefixes.
- The session is not torn down and traffic is not disrupted.
- This is the recommended Soft Reset inbound method.

Outbound method: Soft Reset

- This method simply involves the BGP device sending a new list of prefixes in BGP Updates to the designated neighbor(s).
- The session is not torn down and there is no traffic disruption.

Hard resetting a BGP session



- Command: clear bgp ipv6

Use

- Completely clear the BGP table for all prefixes advertised by a peer.
- Terminate and re-establish a BGP peering to the specified neighbor(s).

Note: Route refresh is the preferred method, when softly resetting a session.

Objective

Hard reset the BGP peering session.

1. Execute a hard reset of the BGP session.

```
BGP_SW1#clear bgp ipv6 2001:db8:b::1
```

Note: The command clear bgp ipv6 * can be used to clear all neighbor sessions.

Verify by viewing the:

- BGP session summary, (note the Up/Down time).
- BGP session to the specified neighbor, (note the counters for the session, the last reset time and notification error message).
- Debug for BGP, (note the Finite State Machine, [FSM] State Change messages. Also note the session is torn down, re-established and new prefixes sent and received).

```
BGP_SW1#show bgp ipv6 summary
BGP router identifier 3.3.3.1, local AS number 300
BGP table version is 53
3 BGP AS-PATH entries
0 BGP community entries

Neighbor        V         AS   MsgRc  MsgSnt TblVer InOutQ Up/Down State/PfxRcd
2001:db8:b::1   4        200     124     131     53 0/0    00:03:42        18
```

```
BGP_SW1#show bgp ipv6 neighbors 2001:db8:b::1
BGP neighbor is 2001:db8:b::1, remote AS 200, local AS 300, external link
  BGP version 4, remote router ID 2.2.2.1
  BGP state = Established, up for 00:03:54
  Last read 00:03:54, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    4-Octet ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 125 messages, 0 notifications, 0 in queue
  Sent 131 messages, 1 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
  Update source is lo
 For address family: IPv6 Unicast
  BGP table version 54, neighbor version 53
  Index 2, Offset 0, Mask 0x4
  Community attribute sent to this neighbor (both)
  18 accepted prefixes
  7 announced prefixes

 Connections established 2; dropped 1
  External BGP neighbor may be up to 2 hops away.
Local host: 2001:db8:c::1, Local port: 54721
Foreign host: 2001:db8:b::1, Foreign port: 179
Nexthop: 3.3.3.1
Nexthop global: 2001:db8:c::1
Nexthop local: ::
BGP connection: non shared network
Last Reset: 00:03:54, due to BGP Notification sent
Notification Error Message: (Cease/Other Configuration Change.)
```

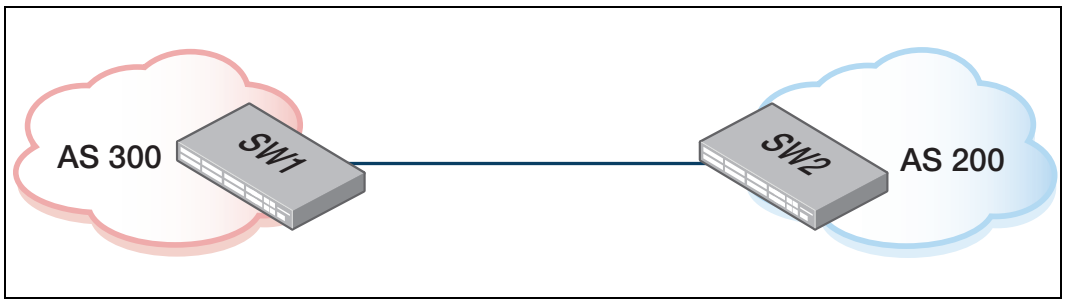
```
BGP_SW1#clear bgp ipv6 2001:db8:b::1
11:11:26 SW1 IMISH[12867]: clear bgp ipv6 2001:db8:b::1
C1#11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State: Established
Event: 35
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Msg-Hdr: Type 3
11:11:26 SW1 BGP[1718]: % BGP-3-NOTIFICATION: sending to 2001:db8:b::1 6/6
(Cease/Other Configuration Change.) 0 data-bytes
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State Change:
Established(6)->Idle(1)
11:11:26 SW1 BGP[1718]: [SFL] route count decreased to 24
11:11:26 SW1 BGP[1718]: [SFL] route count decreased to 23
11:11:26 SW1 BGP[1718]: [SFL] route count decreased to 22
11:11:26 SW1 BGP[1718]: [SFL] route count decreased to 21
11:11:26 SW1 BGP[1718]: [SFL] route count decreased to 20
11:11:26 SW1 BGP[1718]: [SFL] route count decreased to 19
11:11:26 SW1 BGP[1718]: [SFL] route count decreased to 18
11:11:26 SW1 BGP[1718]: [SFL] route count decreased to 17
11:11:26 SW1 BGP[1718]: [SFL] route count decreased to 16
11:11:26 SW1 BGP[1718]: [SFL] route count decreased to 15
11:11:26 SW1 BGP[1718]: [SFL] route count decreased to 14
11:11:26 SW1 BGP[1718]: [SFL] route count decreased to 13
11:11:26 SW1 BGP[1718]: [SFL] route count decreased to 12
[Continued on next page]
```

```
11:11:26 SW1 BGP[1718]: [SFL] route count decreased to 11
11:11:26 SW1 BGP[1718]: [SFL] route count decreased to 10
11:11:26 SW1 BGP[1718]: [SFL] route count decreased to 9
11:11:26 SW1 BGP[1718]: [SFL] route count decreased to 8
11:11:26 SW1 BGP[1718]: [SFL] route count decreased to 7
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] Auto-Start Timer Expiry
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State: Idle Event: 3
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State Change: Idle(1)-
>Connect(2)
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [NETWORK] FD=14, Sock Status: 0-
Success
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State: Connect Event: 17
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Msg-Hdr: Type 1
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Open: Ver 4 MyAS 300
Holdtime 90
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Open: Msg-Size 61
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State Change: Connect(2)-
>OpenSent(4)
11:11:26 SW1 BGP[1718]: [NETWORK] Accept Thread: Incoming conn from host
2001:db8:b::1 (FD=15)
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State: OpenSent Event: 14
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] InConnReq: Tracking...
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Incoming [FSM] InConnReq: Clone creation
successful
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Incoming [NETWORK] FD=15, Sock Status: 0-
Success
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Incoming [FSM] State: Active Event: 17
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Incoming [ENCODE] Msg-Hdr: Type 1
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Incoming [ENCODE] Open: Ver 4 MyAS 300
Holdtime 90
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Incoming [ENCODE] Open: Msg-Size 61
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Incoming [FSM] State Change: Active(3)-
>OpenSent(4)
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Incoming [NETWORK] FD=15, Sock Status:
107-Transport endpoint is not connected
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Incoming [FSM] State: OpenSent Event: 18
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Incoming [FSM] State Change: OpenSent(4)-
>Active(3)
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: type 1, length
61
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: Requesting
immediate Read (42)
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Open: Optional param len
32
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Open Opt: Option Type
2, Option Len 6
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Open Cap: Cap Code 1,
Cap Len 4
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Open Opt: Option Type
2, Option Len 6
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Open Cap: Cap Code 1,
Cap Len 4
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Open Opt: Option Type
2, Option Len 2
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Open Cap: Cap Code 128,
Cap Len 0
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Open Cap: RR Cap(old)
for all address-families
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Open Opt: Option Type
2, Option Len 2
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Open Cap: Cap Code 2,
Cap Len 0
[Continued on next page]
```

```
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Open Cap: RR Cap(new)
for all address-families
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Open Opt: Option Type
2, Option Len 6
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: Requesting
immediate Read (0)
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] KAlive: Received!
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Open Cap: Cap Code 65,
Cap Len 4
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State: OpenSent Event: 19
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Msg-Hdr: Type 4
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Keepalive: 110 KAlive
msg(s) sent
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State Change: OpenSent(4)-
>OpenConfirm(5)
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: type 4, length
19
11:11:26 SW1BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: Requesting
immediate Read (0)
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] KAlive: Received!
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State: OpenConfirm Event:
26
11:11:26 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State Change:
OpenConfirm(5)->Established(6)
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] AS-Origination Timer Expiry

11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] Routeadv Timer Expiry)
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State: Established Event:
33
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State: Established Event:
34
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Msg-Hdr: Type 2
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:3::/64
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:c:20::/64
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:c:10::/64
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:c::1/128
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:4::/64
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update: AFI/SAFI (2/1)
Tot-attr-len 91
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update: Msg #9 Size 114
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Msg-Hdr: Type 2
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:c:30::/64
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:c:40::/64
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update: AFI/SAFI (2/1)
Tot-attr-len 56
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update: Msg #10 Size 79
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: type 2, length
138
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: Requesting
immediate Read (119)
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Update: Starting UPDATE
decoding... Bytes To Read (119), msg_size (119)
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State: Established Event:
27
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix ::/
0
11:11:27 SW1 BGP[1718]: [SFL] route count increased to 8
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:70::/64
11:11:27 SW1 BGP[1718]: [SFL] route count increased to 9

[Continued on next page]
```

```
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:60::/64
 11:11:27 SW1 BGP[1718]: [SFL] route count increased to 10
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:50::/64
11:11:27 SW1 BGP[1718]: [SFL] route count increased to 11
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:40::/64
11:11:27 SW1 BGP[1718]: [SFL] route count increased to 12
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:30::/64
11:11:27 SW1 BGP[1718]: [SFL] route count increased to 13
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:20::/64
11:11:27 SW1 BGP[1718]: [SFL] route count increased to 14
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:10::/64
11:11:27 SW1 BGP[1718]: [SFL] route count increased to 15
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:6::/64
11:11:27 SW1 BGP[1718]: [SFL] route count increased to 16
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: type 2, length
132
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: Requesting
immediate Read (113)
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Update: Starting UPDATE
decoding... Bytes To Read (199), msg_size (113)
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Update: Requesting
immediate Read (19)
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: type 2, length
86
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: Requesting
immediate Read (67)
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Update: Starting UPDATE
decoding... Bytes To Read (67), msg_size (67)
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State: Established Event:
27
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:1::/64
11:11:27 SW1 BGP[1718]: [SFL] route count increased to 17
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:b:20::/64
11:11:27 SW1 BGP[1718]: [SFL] route count increased to 18
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:b:10::/64
11:11:27 SW1 BGP[1718]: [SFL] route count increased to 19
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:b::1/128
11:11:27 SW1 BGP[1718]: [SFL] route count increased to 20
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:4::/64
11:11:27 SW1 BGP[1718]: [SFL] route count increased to 21
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:3::/64
11:11:27 SW1 BGP[1718]: [SFL] route count increased to 22
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:2::/64
11:11:27 SW1 BGP[1718]: [SFL] route count increased to 23
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State: Established Event:
27
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:b:30::/64
11:11:27 SW1 BGP[1718]: [SFL] route count increased to 24
11:11:27 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:b:40::/64
11:11:27 SW1 BGP[1718]: [SFL] route count increased to 25
```

Soft resetting a BGP neighbor session - inbound



- Command: clear bgp ipv6

Use

- Route Refresh softly resets the session by first sending a ROUTE-REFRESH message to the neighbor.
- The neighbor responds by re-sending its prefixes in BGP Update messages.
- The session is not torn down.

Note: Route Refresh is the preferred method, when softly resetting a session.

Objective

Use Route Refresh to softly reset a session to a neighbor inbound.

1. Execute a soft session reset using Route Refresh.

- Notes:
  - Either one of the following commands will perform the same action.
  - Due to RFC 2918, the clear bgp ipv6 { * | neighbor} in command will execute a soft reset and not a hard reset.
  - Hard resets can only be executed using clear bgp ipv6 { * | neighbor} without the added in / out parameter.

```
SW1#clear bgp ipv6 2001:db8:b::1 in
```

or

```
SW1#clear bgp ipv6 2001:db8:b::1 soft in
```

Verify by viewing the:

- BGP prefixes received, (look at the "Last Update" time).
- Neighbor session information on the sending and receiving device, (look at the route refresh counter).
- BGP debug, (note the [ENCODE] Route-Refresh message, and the prefixes received soon after).

```
SW1#show bgp ipv6 2001:db8:a:10::/64
BGP routing table entry for 2001:db8:a:10::/64
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  200 100
    2001:db8:b::1 from 2001:db8:b::1 (2.2.2.1)
      Origin IGP metric 0, localpref 100, valid, external, best
      Last update: Wed Jun  5 09:57:24 2013
```

```
SW1#show bgp ipv6 neighbors 2001:db8:b::1
BGP neighbor is 2001:db8:b::1, remote AS 200, local AS 300, external link
  BGP version 4, remote router ID 2.2.2.1
  BGP state = Established, up for 15:08:13
  Last read 15:08:13, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    4-Octet ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 2138 messages, 0 notifications, 0 in queue
  Sent 2132 messages, 0 notifications, 0 in queue
  Route refresh request: received 0, sent 4
  Minimum time between advertisement runs is 30 seconds
  Update source is lo
 For address family: IPv6 Unicast
  BGP table version 910, neighbor version 909
  Index 2, Offset 0, Mask 0x4
  Community attribute sent to this neighbor (both)
  18 accepted prefixes
  8 announced prefixes

 Connections established 1; dropped 0
  External BGP neighbor may be up to 2 hops away.
Local host: 2001:db8:c::1, Local port: 37381
Foreign host: 2001:db8:b::1, Foreign port: 179
Nexthop: 3.3.3.1
Nexthop global: 2001:db8:c::1
Nexthop local: ::
BGP connection: non shared network
```

```
SW2#show bgp ipv6 neighbors 2001:db8:c::1
BGP neighbor is 2001:db8:c::1, remote AS 300, local AS 200, external link
  BGP version 4, remote router ID 3.3.3.1
  BGP state = Established, up for 15:09:53
  Last read 15:09:53, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    4-Octet ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 2136 messages, 0 notifications, 0 in queue
  Sent 2141 messages, 0 notifications, 0 in queue
  Route refresh request: received 4, sent 0
  Minimum time between advertisement runs is 30 seconds
  Update source is lo
 For address family: IPv6 Unicast
  BGP table version 912, neighbor version 912
  Index 2, Offset 0, Mask 0x4
  Community attribute sent to this neighbor (both)
  8 accepted prefixes
  18 announced prefixes

 Connections established 1; dropped 0
  External BGP neighbor may be up to 2 hops away.
Local host: 2001:db8:b::1, Local port: 179
Foreign host: 2001:db8:c::1, Foreign port: 37381
Nexthop: 2.2.2.1
Nexthop global: 2001:db8:b::1
Nexthop local: ::
BGP connection: non shared network

SW1#clear bgp ipv6 2001:db8:b::1 in
09:57:03 SW1 IMISH[20222]: clear bgp ipv6 2001:db8:b::1 in
SW1#09:57:03 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Msg-Hdr: Type 5
09:57:03 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Route-Refresh: RR-Type
New MsgSize 23
09:57:04 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] Routeadv Timer Expiry)
09:57:04 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State: Established Event:
34
09:57:06 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] AS-Origination Timer Expiry
09:57:06 SW1 BGP[1718]: [RIB] Scanning BGP Network Routes...
09:57:06 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State: Established Event:
33
09:57:20 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: type 4, length
19
09:57:20 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: Requesting
immediate Read (0)
09:57:20 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] KAlive: Received!
09:57:20 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State: Established Event:
26
09:57:21 SW1 BGP[1718]: [RIB] Scanning BGP Network Routes...
09:57:21 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] AS-Origination Timer Expiry
09:57:21 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State: Established Event:
33
09:57:24 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: type 2, length
137
```

[Continued on next page...]

```
09:57:24 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: Requesting
immediate Read (118)
09:57:24 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [DECODE] Update: Starting UPDATE
decoding... Bytes To Read (118), msg_size (118)
09:57:24 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State: Established Event:
27
09:57:24 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:6::/64
09:57:24 SW1 BGP[1718]: [DAMP] bgp_rfd_rt_update(): Route State: NONE, ret=0
09:57:24 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: ...duplicate route
ignored
09:57:24 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:70::/64
09:57:24 SW1 BGP[1718]: [DAMP] bgp_rfd_rt_update(): Route State: NONE, ret=0
09:57:24 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: ...duplicate route
ignored
09:57:24 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:60::/64
09:57:24 SW1 BGP[1718]: [DAMP] bgp_rfd_rt_update(): Route State: NONE, ret=0
09:57:24 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: ...duplicate route
ignored
09:57:24 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:50::/64
09:57:24 SW1 BGP[1718]: [DAMP] bgp_rfd_rt_update(): Route State: NONE, ret=0
09:57:24 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: ...duplicate route
ignored
09:57:24 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:40::/64
09:57:24 SW1 BGP[1718]: [DAMP] bgp_rfd_rt_update(): Route State: NONE, ret=0
09:57:24 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: ...duplicate route
ignored
09:57:24 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:30::/64
09:57:24 SW1 BGP[1718]: [DAMP] bgp_rfd_rt_update(): Route State: NONE, ret=0
09:57:24 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: ...duplicate route
ignored
09:57:24 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:20::/64
09:57:24 SW1 BGP[1718]: [DAMP] bgp_rfd_rt_update(): Route State: NONE, ret=0
09:57:24 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: ...duplicate route
ignored
09:57:24 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:10::/64
```

Soft resetting a BGP neighbor session - outbound

- Command: clear bgp ipv6

Use

- Soft reset a session outbound.
- The device resetting softly outbound advertises its prefixes in BGP Updates without tearing down the session.

Objective

Use soft reset outbound to softly reset a session to a neighbor.

1. **Execute a soft reset outbound.**

Note:

- Either one of these commands will perform the same action.
- Hard resets can only be executed using the command clear bgp ipv6 { * | neighbor} without the added in / out parameter.

```
SW1#clear bgp ipv6 2001:db8:b::1 out
```

OR

```
SW1#clear bgp ipv6 2001:db8:b::1 soft out
```

Verify by viewing the:

- BGP prefixes received on SW2, (look at the "Last Update" time).
- BGP debug on the sending device, (note the [ENCODE] Update MP Reach messages).

```
SW2#show bgp ipv6 2001:db8:c:10::/64
BGP routing table entry for 2001:db8:c:10::/64
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
  2001:db8:a::1
  300
    2001:db8:c::1 from 2001:db8:c::1 (3.3.3.1)
      Origin incomplete metric 0, localpref 100, valid, external, best
      Last update: Wed Jun  5 09:36:59 2013
```

```
SW1#clear bgp ipv6 2001:db8:b::1 soft out
10:36:52 SW1 IMISH[12867]: clear bgp ipv6 2001:db8:b::1 soft out
10:36:53 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] AS-Origination Timer Expiry

10:36:53 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State: Established Event:
33
10:36:58 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] Keep-alive-Timer Expiry
10:36:58 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State: Established Event:
11
10:36:58 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Msg-Hdr: Type 4
10:36:58 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Keepalive: 28 KAlive
msg(s) sent
10:36:58 SW1 BGP[1718]: [RIB] Scanning BGP Network Routes...
10:36:59 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] Routeadv Timer Expiry)
10:36:59 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [FSM] State: Established Event:
34
10:36:59 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Msg-Hdr: Type 2
10:36:59 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:3::/64
10:36:59 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:c:20::/64
10:36:59 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:c:10::/64
10:36:59 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:c::1/128
10:36:59 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:4::/64
10:36:59 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update: AFI/SAFI (2/1)
Tot-attr-len 91
10:36:59 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update: Msg #7 Size 114
10:36:59 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Msg-Hdr: Type 2
10:36:59 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:c:30::/64
10:36:59 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:c:40::/64
10:36:59 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update: AFI/SAFI (2/1)
Tot-attr-len 56
10:36:59 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [ENCODE] Update: Msg #8 Size 79
```

Performing a "soft" update when Route Refresh is not supported on a BGP4+ peer

- Command parameter: soft-reconfiguration inbound

Use

- When a BGP neighbor does not support the "Route Refresh" message, soft reconfiguration is used. The command can be used to allow soft updates, instead of tearing the existing session down and establishing a new session to receive new prefix information.
- The soft-reconfiguration inbound command causes the BGP4+ router to retain an unfiltered table of the received prefixes. This is stored in memory on the local device. This is separate from the BGP table.
- The clear bgp ipv6 { * | neighbor address} soft-in command, causes the router to use the unfiltered table in memory to generate new inbound updates, locally on the device. The newly generated prefixes are installed in the BGP table.
- Clearing a session using the soft reconfiguration method avoids the BGP session being reset.

Note:

- All AlliedWare Plus switches that support BGP4+ are capable of Route Refresh. This scenario is used as an example only.
- Route Refresh is recommended if available for BGP4+ peers.
- Configuring soft reconfiguration will take precedence over the route refresh capability for the specified neighbor, therefore it is recommended engineers carefully consider if this method is required when route refresh is available.
- This method can be memory intensive.

Objective

Use the soft reconfiguration method in this simulation where Route Refresh is not used.

- Because route refresh is not supported on the neighbor, we cannot receive a new list of prefixes without the session being torn down.
- Soft reconfiguration will be configured on SW1 so prefixes received from the BGP peer neighbor will be stored.

- When the session needs to be cleared, we can do this without tearing down the peering session and interrupting traffic.

1. Setup soft-recognition inbound on SW1.

```
SW1(config)#router bgp 300
SW1(config-router)#address-family ipv6
SW1(config-router-af)#neighbor 2001:db8:b::1 soft-reconfiguration
   inbound
```

2. Perform a soft reset on the BGP neighbor connection.

```
SW1#clear bgp ipv6 2001:db8:b::1 soft in
Or
SW1#clear bgp ipv6 2001:db8:b::1 in
```

Verify by viewing the:

- **show bgp ipv6 neighbors received-routes** to view an unfiltered list of all routes received from the peer.
- Prefixes received when updates were generated from the stored list. (Note the last update time).
- Neighbor session information. Notice the Inbound soft reconfiguration is allowed (IPv6 address family). (Also note that because we are using soft reconfiguration, the route refresh counters do not increment).

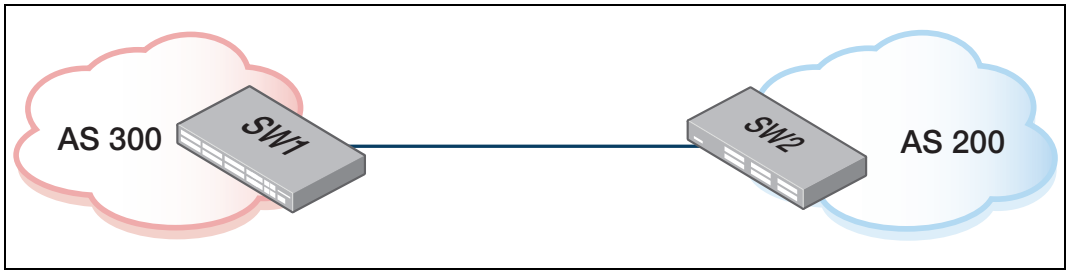
Optional: Enable debug of BGP updates to see the immediate soft refresh, using the stored prefixes.

```
SW1#show bgp ipv6 2001:db8:a:10::/64
BGP routing table entry for 2001:db8:a:10::/64
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  200 100
    2001:db8:b::1 from 2001:db8:b::1 (2.2.2.1)
      Origin IGP metric 0, localpref 100, valid, external, best
      Last update: Wed Jun  4 18:54:51 2013
```

```
SW1#show bgp ipv6 neighbors 2001:db8:b::1
16:26:04 C1 IMISH[8960]: show bgp ipv6 neighbors 2001:db8:b::1
BGP neighbor is 2001:db8:b::1, remote AS 200, local AS 300, external link
  BGP version 4, remote router ID 2.2.2.1
  BGP state = Established, up for 00:04:20
  Last read 00:04:20, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    4-Octet ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 146 messages, 1 notifications, 0 in queue
  Sent 142 messages, 2 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
  Update source is lo

For address family: IPv6 Unicast
  BGP table version 6, neighbor version 6
  Index 2, Offset 0, Mask 0x4
  Inbound soft reconfiguration allowed
  Community attribute sent to this neighbor (both)
  18 accepted prefixes
  8 announced prefixes
Connections established 4; dropped 3
  External BGP neighbor may be up to 2 hops away.
Local host: 2001:db8:c::1, Local port: 179
Foreign host: 2001:db8:b::1, Foreign port: 35476
Nexthop: 3.3.3.1
Nexthop global: 2001:db8:c::1
Nexthop local: ::
BGP connection: non shared network
Last Reset: 00:04:20, due to BGP Notification sent
Notification Error Message: (Cease/Other Configuration Change.)]
```

```
SW1#clear bgp ipv6 2001:db8:b::1 in
18:54:51 SW1 IMISH[4010]: clear bgp ipv6 2001:db8:b::1 in
18:54:51 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix ::/
0
18:54:51 SW1 BGP[1718]: [DAMP] bgp_rfd_rt_update(): Route State: NONE, ret=0
18:54:51 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: ...duplicate route
ignored
18:54:51 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:1::/64
18:54:51 SW1 BGP[1718]: [DAMP] bgp_rfd_rt_update(): Route State: NONE, ret=0
18:54:51 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: ...duplicate route
ignored
18:54:51 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:2::/64
18:54:51 SW1 BGP[1718]: [DAMP] bgp_rfd_rt_update(): Route State: NONE, ret=0
18:54:51 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: ...duplicate route
ignored
18:54:51 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:3::/64
18:54:51 SW1 BGP[1718]: [DAMP] bgp_rfd_rt_update(): Route State: NONE, ret=0
18:54:51 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: ...duplicate route
ignored
18:54:51 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:6::/64
18:54:51 SW1 BGP[1718]: [DAMP] bgp_rfd_rt_update(): Route State: NONE, ret=0
18:54:51 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: ...duplicate route
ignored
18:54:51 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:10::/64
18:54:51 SW1 BGP[1718]: [DAMP] bgp_rfd_rt_update(): Route State: NONE, ret=0
18:54:51 SW1 BGP[1718]: 2001:db8:b::1-Outgoing [RIB] Update: ...duplicate route
ignored
```

Preventing termination of a BGP session when capability negotiation issues occur



- Command parameter: dont-capability-negotiate

Function

The dont-capability-negotiate command, prevents negotiation of BGP capabilities on a BGP speaker by suppressing the *Capabilities Optional Parameter* in the OPEN message, (Optional Capabilities are removed from the OPEN message).

When to use this feature:

- This feature can be used when a BGP peer does not support capabilities the speaker is advertising and negotiation issues occur.
- Negotiation issues may be either the session terminates and re-establishes or will terminate and require manual intervention to re-establish.

For more information go to:

http://tools.ietf.org/html/rfc5492

Objective:

- Prevent termination of BGP session when capability negotiation issues occur.
- In this simulation, the BGP peer was an AlliedWare Plus device that was setup to not advertise capabilities to SW1.
- SW1 is advertising to SW2 that it supports the graceful-restart capability.
- A mismatch occurs every time the session initiates for the first time, and when the session is cleared.

For an example without dont-capability negotiate, see Example without "dont-capability-negotiate" on page 232

1. Configure **dont-capability-negotiate.**

```
SW1(config)#router bgp 300
SW1(config-router)#neighbor 2001:db8:b::1 dont-capability-negotiate
```

2. Clear the BGP session.

```
SW1#clear bgp ipv6 2001:db8:b::1
```

Verify by viewing the:

- Neighbor session. (*Unsupported Optional Parameter* of *Unsupported Capability* message is not present.)
- Debug:
  - notice the OPEN message sent – look for "Outgoing [ENCODE] Open:" - no optional capabilities are advertised.
  - notice there are no NOTIFICATION negotiation errors or session terminations.
- View the OPEN message by sniffing the LAN traffic via a network protocol analyzer tool. There are no optional capabilities advertised to the peer.

```
SW1#show bgp ipv6 neighbors 2001:db8:b::1
BGP neighbor is 2001:db8:b::1, remote AS 200, local AS 300, external link
  BGP version 4, remote router ID 2.2.2.1
  BGP state = Established, up for 00:12:47
  Last read 00:12:47, hold time is 90, keepalive interval is 30 seconds
  Received 2744 messages, 6 notifications, 0 in queue
  Sent 2734 messages, 12 notifications, 0 in queue
   Route refresh request: received 0, sent 0
   4-Octet ASN Capability: advertised and received
  Minimum time between advertisement runs is 30 seconds
  Update source is lo
  prefixes
For address family: IPv6 Unicast
  BGP table version 1162, neighbor version 1162
  Index 2, Offset 0, Mask 0x4
  AF-dependant capabilities:
  Community attribute sent to this neighbor (both)
  18 accepted prefixes
  7 announced prefixes

 Connections established 13; dropped 12
  External BGP neighbor may be up to 2 hops away.
Local host: 2001:db8:c::1, Local port: 44125
Foreign host: 2001:db8:b::1, Foreign port: 179
Nexthop: 3.3.3.1
Nexthop global: 2001:db8:c::1
Nexthop local: ::
BGP connection: non shared network
Last Reset: 00:12:47, due to BGP Notification sent
Notification Error Message: (Cease/Other Configuration Change.)
```

```
SW1#clear bgp ipv6 2001:db8:b::1
11:12:34 SW1 IMISH[19834]: clear bgp ipv6 2001:db8:b::1
SW1#11:12:34 C1 BGP[1716]: 2001:db8:b::1-Outgoing [FSM] State: Established
Event: 35
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [ENCODE] Msg-Hdr: Type 3
11:12:34 SW1 BGP[1716]: % BGP-3-NOTIFICATION: sending to 2001:db8:b::1 6/6
(Cease/Other Configuration Change.) 0 data-bytes
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [FSM] State Change:
Established(6)->Idle(1)
11:12:34 SW1 BGP[1716]: [SFL] route count decreased to 24
11:12:34 SW1 BGP[1716]: [SFL] route count decreased to 23
11:12:34 SW1 BGP[1716]: [SFL] route count decreased to 22
11:12:34 SW1 BGP[1716]: [SFL] route count decreased to 21
11:12:34 SW1 BGP[1716]: [SFL] route count decreased to 20
11:12:34 SW1 BGP[1716]: [SFL] route count decreased to 19
11:12:34 SW1 BGP[1716]: [SFL] route count decreased to 18
11:12:34 SW1 BGP[1716]: [SFL] route count decreased to 17
11:12:34 SW1 BGP[1716]: [SFL] route count decreased to 16
11:12:34 SW1 BGP[1716]: [SFL] route count decreased to 15
11:12:34 SW1 BGP[1716]: [SFL] route count decreased to 14
11:12:34 SW1 BGP[1716]: [SFL] route count decreased to 13
11:12:34 SW1 BGP[1716]: [SFL] route count decreased to 12
11:12:34 SW1 BGP[1716]: [SFL] route count decreased to 11
11:12:34 SW1 BGP[1716]: [SFL] route count decreased to 10
11:12:34 SW1 BGP[1716]: [SFL] route count decreased to 9
11:12:34 SW1 BGP[1716]: [SFL] route count decreased to 8
11:12:34 SW1 BGP[1716]: [SFL] route count decreased to 7
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [FSM] Auto-Start Timer Expiry
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [FSM] State: Idle Event: 3
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [FSM] State Change: Idle(1)-
>Connect(2)
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [NETWORK] FD=14, Sock Status: 0-
Success
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [FSM] State: Connect Event: 17
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [ENCODE] Msg-Hdr: Type 1
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [ENCODE] Open: Ver 4 MyAS 300
Holdtime 90
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [ENCODE] Open: Msg-Size 29
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [FSM] State Change: Connect(2)-
>OpenSent(4)
11:12:34 SW1 BGP[1716]: [NETWORK] Accept Thread: Incoming conn from host
2001:db8:b::1 (FD=15)
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: type 1, length
29
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: Requesting
immediate Read (10)
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [DECODE] Open: Optional param len
0
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [FSM] State: OpenSent Event: 14
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [FSM] InConnReq: Tracking...
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Incoming [FSM] InConnReq: Clone creation
successful
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Incoming [NETWORK] FD=15, Sock Status: 0-
Success
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [FSM] State: OpenSent Event: 19
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [ENCODE] Msg-Hdr: Type 4
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [ENCODE] Keepalive: 2649 KAlive
msg(s) sent
```

[Continued on next page...]

```
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [FSM] State Change: OpenSent(4)-
>OpenConfirm(5)
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Incoming [FSM] State: Active Event: 17
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Incoming [ENCODE] Msg-Hdr: Type 1
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Incoming [ENCODE] Open: Ver 4 MyAS 300
Holdtime 90
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Incoming [ENCODE] Open: Msg-Size 29
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Incoming [FSM] State Change: Active(3)-
>OpenSent(4)
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Incoming [NETWORK] FD=15, Sock Status:
107-Transport endpoint is not connected
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Incoming [FSM] State: OpenSent Event: 18
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Incoming [FSM] State Change: OpenSent(4)-
>Active(3)
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: type 4, length
19
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: Requesting
immediate Read (0)
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [DECODE] KAlive: Received!
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [FSM] State: OpenConfirm Event:
26
11:12:34 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [FSM] State Change:
OpenConfirm(5)->Established(6)
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [FSM] AS-Origination Timer Expiry
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [FSM] Routeadv Timer Expiry)
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [FSM] State: Established Event:
33
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [FSM] State: Established Event:
34
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [ENCODE] Msg-Hdr: Type 2
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:3::/64
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:c:20::/64
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:c:10::/64
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:c::1/128
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:4::/64
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [ENCODE] Update: AFI/SAFI (2/1)
Tot-attr-len 89
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [ENCODE] Update: Msg #29 Size 112
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [ENCODE] Msg-Hdr: Type 2
11:12:35 SW1SW1 BGP[1716]: 2001:db8:b::1-Outgoing [ENCODE] Update MP Reach:
Prefix 2001:db8:c:30::/64
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:c:40::/64
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [ENCODE] Update: AFI/SAFI (2/1)
Tot-attr-len 54
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [ENCODE] Update: Msg #30 Size 77
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: type 2, length
134
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: Requesting
immediate Read (115)
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [DECODE] Update: Starting UPDATE
decoding... Bytes To Read (115), msg_size (115)
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [FSM] State: Established Event:
27
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix ::/
0
11:12:35 SW1 BGP[1716]: [SFL] route count increased to 8
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:70::/64
```

[Continued on next page...]

```
11:12:35 SW1 BGP[1716]: [SFL] route count increased to 9
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:60::/64
11:12:35 SW1 BGP[1716]: [SFL] route count increased to 10
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:50::/64
11:12:35 SW1 BGP[1716]: [SFL] route count increased to 11
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:40::/64
11:12:35 SW1 BGP[1716]: [SFL] route count increased to 12
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:30::/64
11:12:35 SW1 BGP[1716]: [SFL] route count increased to 13
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:20::/64
11:12:35 SW1 BGP[1716]: [SFL] route count increased to 14
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:a:10::/64
11:12:35 SW1 BGP[1716]: [SFL] route count increased to 15
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:6::/64
11:12:35 SW1 BGP[1716]: [SFL] route count increased to 16
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: type 2, length 130
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: Requesting immediate
Read (111)
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [DECODE] Update: Starting UPDATE
decoding... Bytes To Read (195), msg_size (111)
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [DECODE] Update: Requesting immediate Read
(19)
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: type 2, length 84
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [DECODE] Msg-Hdr: Requesting immediate
Read (65)
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [DECODE] Update: Starting UPDATE
decoding... Bytes To Read (65), msg_size (65)
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [FSM] State: Established Event: 27
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:1::/64
11:12:35 SW1 BGP[1716]: [SFL] route count increased to 17
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:b:20::/64
11:12:35 SW1 BGP[1716]: [SFL] route count increased to 18
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:b:10::/64
11:12:35 SW1 BGP[1716]: [SFL] route count increased to 19
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:b::1/128
11:12:35 SW1 BGP[1716]: [SFL] route count increased to 20
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:4::/64
11:12:35 SW1 BGP[1716]: [SFL] route count increased to 21
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:3::/64
11:12:35 SW1 BGP[1716]: [SFL] route count increased to 22
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:2::/64
11:12:35 SW1 BGP[1716]: [SFL] route count increased to 23
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [FSM] State: Established Event: 27
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:b:30::/64
11:12:35 SW1 BGP[1716]: [SFL] route count increased to 24
11:12:35 SW1 BGP[1716]: 2001:db8:b::1-Outgoing [RIB] Update: Received Prefix
2001:db8:b:40::/64
11:12:35 SW1 BGP[1716]: [SFL] route count increased to 25
11:12:44 SW1 BGP[1716]: [RIB] Scanning BGP Network Routes...
```

Example without "dont-capability-negotiate"

Notice the *Notification Error Message*

```
SW1#show bgp ipv6 neighbors 2001:db8:b::1
BGP neighbor is 2001:db8:b::1, remote AS 300, local AS 200, external link
  BGP version 4, remote router ID 3.3.3.1
  BGP state = Established, up for 00:02:18
  Last read 00:02:18, hold time is 90, keepalive interval is 30 seconds
  Received 2762 messages, 7 notifications, 0 in queue
  Sent 2752 messages, 13 notifications, 0 in queue
    Route refresh request: received 0, sent 0
    4-Octet ASN Capability: advertised and received
  Minimum time between advertisement runs is 30 seconds
  Update source is lo
For address family: IPv6 Unicast
  BGP table version 1169, neighbor version 1169
  Index 2, Offset 0, Mask 0x4
  AF-dependant capabilities:
  Community attribute sent to this neighbor (both)
  18 accepted prefixes
  7 announced prefixes

 Connections established 14; dropped 13
  External BGP neighbor may be up to 2 hops away.
Local host: 2001:db8:b::1, Local port: 179
Foreign host: 2001:db8:c::1, Foreign port: 50379
Nexthop: 2.2.2.1
Nexthop global: 2001:db8:b::1
Nexthop local: ::
BGP connection: non shared network
Last Reset: 00:02:18, due to BGP Notification received
Notification Error Message: (OPEN Message Error/Unsupported Optional Parameter.)
```

Debug example when dont-capability-negotiate is not configured and unsupported optional parameters are advertised by the speaker

Note:

- The OPEN message size sent is 63 bytes instead of 29 bytes. The optional parameters are being advertised by SW1 (the speaker).

```
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Open: Msg-Size 63
```

- SW1 (the speaker) receives an OPEN message with no optional parameters and is a mismatch.

```
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [DECODE] Open: Optional param len 0
```

- SW2 terminates the session and re-establishes the session. This time SW1 advertises an OPEN message without the additional parameters. (In accordance with RFC 5492.)

```
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Msg-Hdr: Type 1
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Open: Ver 4 MyAS 100
Holdtime 90
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Open: Msg-Size 29
```

If dont-capability-negotiate was enabled. This whole process would not be required.

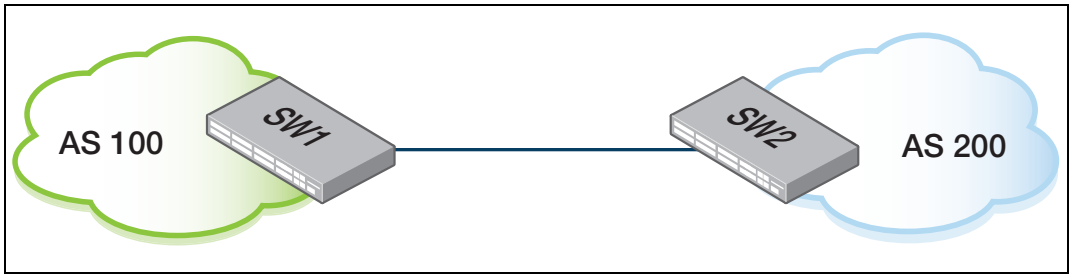
```
14:52:13 A2 BGP[1842]: [NETWORK] Accept Thread: Incoming conn from host
2001:db8:5::2 (FD=14)
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] State: Active Event: 14
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] InConnReq: Accepting...
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [NETWORK] FD=14, Sock Status: 0-
Success
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] State: Active Event: 17
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Msg-Hdr: Type 1
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Open: Ver 4 MyAS 100
Holdtime 90
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Open: Msg-Size 63
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] State Change: Active(3)-
>OpenSent(4)
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [DECODE] Msg-Hdr: type 1, length
29
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [DECODE] Msg-Hdr: Requesting
immediate Read (10)
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [DECODE] Open: Optional param len
0
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] State: OpenSent Event: 19
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Msg-Hdr: Type 4
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Keepalive: 1 KAlive
msg(s) sent
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] State Change: OpenSent(4)-
>OpenConfirm(5)
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [DECODE] Msg-Hdr: type 3, length
21
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [DECODE] Msg-Hdr: Requesting
immediate Read (2)
14:52:13 A2 BGP[1842]: % BGP-3-NOTIFICATION: received from 2001:db8:5::2 2/4
(OPEN Message Error/Unsupported Optional Parameter.) 0 data-bytes
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] State: OpenConfirm Event: 25
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] BGP Notification received
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] State Change:
OpenConfirm(5)->Idle(1)
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] Auto-Start Timer Expiry
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] State: Idle Event: 3
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] State Change: Idle(1)-
>Connect(2)
14:52:13 A2 BGP[1842]: [NETWORK] Accept Thread: Incoming conn from host
2001:db8:5::2 (FD=16)
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] State: Connect Event: 14
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] InConnReq: Accepting...
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [NETWORK] FD=16, Sock Status: 0-
Success
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] State: Connect Event: 17
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Msg-Hdr: Type 1
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Open: Ver 4 MyAS 100
Holdtime 90
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Open: Msg-Size 29
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] State Change: Connect(2)-
>OpenSent(4)
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [DECODE] Msg-Hdr: type 1, length
29
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [DECODE] Msg-Hdr: Requesting
immediate Read (10)
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [DECODE] Open: Optional param len
0
[Continued on next page...]
```

```
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] State: OpenSent Event: 19
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Msg-Hdr: Type 4
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Keepalive: 2 KAlive
msg(s) sent
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] State Change: OpenSent(4)-
>OpenConfirm(5)
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [DECODE] Msg-Hdr: type 4, length
19
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [DECODE] Msg-Hdr: Requesting
immediate Read (0)
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [DECODE] KAlive: Received!
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] State: OpenConfirm Event: 26
14:52:13 A2 BGP[1842]: % BGP-5-ADJCHANGE: neighbor 2001:db8:5::2 Up
14:52:13 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] State Change:
OpenConfirm(5)->Established(6)
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] AS-Origination Timer Expiry
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] Routeadv Timer Expiry)
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] State: Established Event: 33
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] State: Established Event: 34
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Msg-Hdr: Type 2
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update MP Reach: Prefix
::/0
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:a:70::/64
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:5::/64
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update: AFI/SAFI (2/1)
Tot-attr-len 71
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update: Msg #1 Size 94
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Msg-Hdr: Type 2
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8::1/128
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update: AFI/SAFI (2/1)
Tot-attr-len 76
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update: Msg #2 Size 99
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Msg-Hdr: Type 2
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:1::/64
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:2::/64
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update: AFI/SAFI (2/1)
Tot-attr-len 77
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update: Msg #3 Size 100
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Msg-Hdr: Type 2
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:a:10::/64
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:ffff::/64
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:a:30::/64
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update: AFI/SAFI (2/1)
Tot-attr-len 86
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update: Msg #4 Size 109
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Msg-Hdr: Type 2
```

[Continued on next page...]

```
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:a:20::/64
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:a:60::/64
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:a:40::/64
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update: AFI/SAFI (2/1)
Tot-attr-len 86
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update: Msg #5 Size 109
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [FSM] State: Established Event: 34
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Msg-Hdr: Type 2
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:b:10::/64
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:b:40::/64
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:b:30::/64
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update MP Reach: Prefix
2001:db8:b:20::/64
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update: AFI/SAFI (2/1)
Tot-attr-len 95
14:52:14 A2 BGP[1842]: 2001:db8:5::2-Outgoing [ENCODE] Update: Msg #6 Size 118
```

Setup BGP graceful-restart capability for all BGP neighbors



- Command: bgp graceful-restart

BGP Graceful restart function

Graceful restart allows the data forwarding plane to process and forward packets in the event that the control plane fails.

RFC 4724, Section 3 states: "*The Graceful Restart Capability is a new BGP capability [BGP-CAP] that can be used by a BGP speaker to indicate its ability to preserve its forwarding state during BGP restart. It can also be used to convey to its peer its intention of generating the End-of-RIB marker upon the completion of its initial routing updates.*"

Operation overview

1. BGP speaker advertises the Graceful Restart Capability to its peer(s).
2. BGP speaker sends an "End-of-RIB" to its peer(s), upon initial routing update completion.
3. The peers retain their forwarding state for learned BGP prefixes and marks these routes as stale. The BGP peers do not differentiate between the stale routes and other routing information, and continue to use the stale routes when forwarding traffic.
4. Upon re-establishment of sessions to BGP peers, the restarting speaker will set the "Restart State" bit in the OPEN message.
5. BGP peers advertise routing information to the restarted speaker using UPDATE messages followed by an End-of-RIB marker. When all peers have sent their End-of-RIB marker, the restarted BGP speaker processes the new updates.
6. The restarted BGP speaker then performs route selection and updates its routing information, while deleting the older stale entries.
7. Once this process is complete, the restarted BGP speaker may then advertise updates to BGP peers, followed by an End-of-RIB marker.

Graceful-restart End-of-RIB marker for BGP4+

This is a BGP UPDATE message that contains only the MP_UNREACH_NLRI attribute.

Use

- This command globally enables graceful restart for all BGP peers.
- Use this function to minimize the duration of an outage and reduce route flaps, when the BGP process fails.

For more information see: http://tools.ietf.org/html/rfc4724

Other AlliedWare Plus Graceful restart parameters

```
awplus(config-router)#bgp graceful-restart ?
 graceful-reset Graceful-reset capability
 restart-time Max time needed for Neighbor(s) to restart
 stalepath-time Max time to retain stale paths from restarting Neighbor(s)
```

Graceful-reset: Use this parameter to enable graceful restart when a configuration change forces a peer restart, note that graceful-restart must be configured first.

Restart-time: Use this parameter to set the maximum time that the graceful restart BGP speaker will wait for a peer to come back up. This value is applied to neighbors unless it is overridden by configuring the corresponding value on the neighbor.

Stalepath-time: Use this parameter to set the maximum time to preserve stale paths from a gracefully restarted neighbor. All stalepaths, unless reinstated after a re-establishment, will be deleted when the stalepath-time expires.

Objective:

Perform a BGP graceful restart on SW1.

1. Enable BGP graceful restart.

```
SW1(config-router)#bgp graceful-restart
SW2(config-router)#bgp graceful-restart
```

2. Advertise the graceful restart capability to each BGP peer.

Note: Initiating this command will perform a hard reset of the BGP session.

This command advertises the graceful restart capability to a neighbor.

```
SW1(config-router)#address-family ipv6
SW1(config-router-af)#neighbor 2001:db8:b::1 capability graceful-
   restart
SW2(config-router)#address-family ipv6
SW2(config-router-af)#neighbor 2001:db8:a::1 capability graceful-
   restart
```

Verify by:

- Viewing the BGP neighbor information.

  Note the "Graceful restart: advertised, received"

  Note the "Graceful restart status."

- Viewing the routing table on SW1. (Notice the "p" status indicator marking the routes "stale".)

```
SW1#show bgp ipv6 neighbors 2001:db8:b::1
09:45:53 SW1 IMISH[3898]: show bgp ipv6 neighbors 2001:db8:b::1
BGP neighbor is 2001:db8:b::1, remote AS 200, local AS 100, external link
  BGP version 4, remote router ID 2.2.2.1
  BGP state = Established, up for 00:02:05
  Last read 00:02:05, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    4-Octet ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 46 messages, 3 notifications, 0 in queue
  Sent 58 messages, 0 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
  Update source is lo
 For address family: IPv6 Unicast
  BGP table version 21, neighbor version 20
  Index 2, Offset 0, Mask 0x4
  AF-dependant capabilities:
    Graceful restart: advertised, received

  Community attribute sent to this neighbor (both)
  Default information originate, default sent
  8 accepted prefixes
  10 announced prefixes

 Connections established 4; dropped 3
 Graceful-restart Status:
  Remote restart-time is 90 sec

  External BGP neighbor may be up to 2 hops away.
Local host: 2001:db8:a::1, Local port: 179
Foreign host: 2001:db8:b::1, Foreign port: 50765
Nexthop: 1.1.1.1
Nexthop global: 2001:db8:a::1
Nexthop local: ::
BGP connection: non shared network
Last Reset: 00:02:05, due to BGP Notification received
Notification Error Message: (Cease/Other Configuration Change.)
```

- Routes are stale until SW1 has performed route selection and updates the routes.

```
SW1#show ipv6 route database
10:23:39 SW1 IMISH[3898]: show ipv6 route database
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
       IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
       > - selected route, * - FIB route, p - stale info
Timers: Uptime
C  *> 2001:db8:1::/64 via ::, vlan100, 01:12:22
B  *>p2001:db8:3::/64 [20/0] via 2001:db8:b::1 (recursive via fe80::200:cdff:fe29:ec11), vlan100,
00:01:00
B  *>p2001:db8:4::/64 [20/0] via 2001:db8:b::1 (recursive via fe80::200:cdff:fe29:ec11), vlan100,
00:01:00
C  *> 2001:db8:a::1/128 via ::, lo, 01:12:32
O     2001:db8:a:10::/64 [110/1] via ::, vlan10, 01:12:20
C  *> 2001:db8:a:10::/64 via ::, vlan10, 01:12:21
OIA*> 2001:db8:a:20::/64 [110/2] via fe80::209:41ff:fefb:c323, vlan10, 01:11:26
OIA*> 2001:db8:a:30::/64 [110/2] via fe80::209:41ff:fefb:c323, vlan10, 01:11:26
OIA*> 2001:db8:a:40::/64 [110/3] via fe80::209:41ff:fefb:c323, vlan10, 01:11:26
OE2*> 2001:db8:a:50::/64 [110/20] via fe80::209:41ff:fefb:c323, vlan10, 01:11:26
OIA*> 2001:db8:a:60::/64 [110/3] via fe80::209:41ff:fefb:c323, vlan10, 01:11:26
OIA*> 2001:db8:a:70::/64 [110/3] via fe80::209:41ff:fefb:c323, vlan10, 01:11:26
B    p2001:db8:b::1/128 [20/0] via 2001:db8:b::1 inactive, 00:01:00
S  *> 2001:db8:b::1/128 [1/0] via fe80::200:cdff:fe29:ec11, vlan100, 01:12:32
  >                     [1/0] via fe80::200:cdff:fe29:ec11, vlan200 inactive, 01:12:32
B  *>p2001:db8:b:10::/64 [20/0] via 2001:db8:b::1 (recursive via fe80::200:cdff:fe29:ec11),
vlan100, 00:01:00
B  *>p2001:db8:b:20::/64 [20/0] via 2001:db8:b::1 (recursive via fe80::200:cdff:fe29:ec11),
vlan100, 00:01:00
B  *>p2001:db8:b:30::/64 [20/2] via 2001:db8:b::1 (recursive via fe80::200:cdff:fe29:ec11),
vlan100, 00:01:00
B  *>p2001:db8:b:40::/64 [20/2] via 2001:db8:b::1 (recursive via fe80::200:cdff:fe29:ec11),
vlan100, 00:01:00
OE2*> 2001:db8:17:17::/64 [110/20] via fe80::209:41ff:fefb:c323, vlan10, 01:11:26
C  *  fe80::/64 via ::, vlan10, 01:12:21
C  *  fe80::/64 via ::, vlan100, 01:12:22
C  *> fe80::/64 via ::, lo, 01:12:32
```

- New routes have been added as soon as the best paths have been selected.

```
SW1#show ipv6 route database
10:23:40 SW1 IMISH[3898]: show ipv6 route database
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
       IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
       > - selected route, * - FIB route, p - stale info
Timers: Uptime

C  *> 2001:db8:1::/64 via ::, vlan100, 01:12:23
B  *> 2001:db8:3::/64 [20/0] via 2001:db8:b::1 (recursive via fe80::200:cdff:fe29:ec11), vlan100,
00:00:01
B  *> 2001:db8:4::/64 [20/0] via 2001:db8:b::1 (recursive via fe80::200:cdff:fe29:ec11), vlan100,
00:00:01
C  *> 2001:db8:a::1/128 via ::, lo, 01:12:33
O     2001:db8:a:10::/64 [110/1] via ::, vlan10, 01:12:21
C  *> 2001:db8:a:10::/64 via ::, vlan10, 01:12:22
OIA*> 2001:db8:a:20::/64 [110/2] via fe80::209:41ff:fefb:c323, vlan10, 01:11:27
OIA*> 2001:db8:a:30::/64 [110/2] via fe80::209:41ff:fefb:c323, vlan10, 01:11:27
OIA*> 2001:db8:a:40::/64 [110/3] via fe80::209:41ff:fefb:c323, vlan10, 01:11:27
OE2*> 2001:db8:a:50::/64 [110/20] via fe80::209:41ff:fefb:c323, vlan10, 01:11:27
OIA*> 2001:db8:a:60::/64 [110/3] via fe80::209:41ff:fefb:c323, vlan10, 01:11:27
OIA*> 2001:db8:a:70::/64 [110/3] via fe80::209:41ff:fefb:c323, vlan10, 01:11:27
B     2001:db8:b::1/128 [20/0] via 2001:db8:b::1 inactive, 00:00:01
S  *> 2001:db8:b::1/128 [1/0] via fe80::200:cdff:fe29:ec11, vlan100, 01:12:33
  >                     [1/0] via fe80::200:cdff:fe29:ec11, vlan200 inactive, 01:12:33
B  *> 2001:db8:b:10::/64 [20/0] via 2001:db8:b::1 (recursive via fe80::200:cdff:fe29:ec11),
vlan100, 00:00:01
B  *> 2001:db8:b:20::/64 [20/0] via 2001:db8:b::1 (recursive via fe80::200:cdff:fe29:ec11),
vlan100, 00:00:01
B  *> 2001:db8:b:30::/64 [20/2] via 2001:db8:b::1 (recursive via fe80::200:cdff:fe29:ec11),
vlan100, 00:00:01
B  *> 2001:db8:b:40::/64 [20/2] via 2001:db8:b::1 (recursive via fe80::200:cdff:fe29:ec11),
vlan100, 00:00:01
OE2*> 2001:db8:17:17::/64 [110/20] via fe80::209:41ff:fefb:c323, vlan10, 01:11:27
C  *  fe80::/64 via ::, vlan10, 01:12:22
C  *  fe80::/64 via ::, vlan100, 01:12:23
C  *> fe80::/64 via ::, lo, 01:12:33
```

Detailed view of the process

- SW1's View:

```
SW1#restart bgp graceful
09:50:36 SW1 IMISH[3898]: restart bgp graceful

% Cannot get reply from protocol daemon
SW1#09:50:36 SW1 BGP[30060]: SFL: Base feature license allocated
09:50:36 SW1 BGP[30060]: SFL: [bgpd] LicenseCheck: BGP4+ is active
09:50:36 SW1 BGP[30060]: SFL: [bgpd] LicenseCheck: returns Success.
09:50:36 SW1 BGP[30060]: SFL: bgp_cap_have_ipv6 = 1
09:50:36 SW1 BGP[30060]: SFL: [bgpd] LicenseCheck: BGP-FULL is active
09:50:36 SW1 BGP[30060]: SFL: [bgpd] LicenseCheck: returns Success.
09:50:36 SW1 BGP[30060]: AIS deps met for bgpd
09:50:36 SW1 NSM[1864]: nsm_server_recv_service: proto 7 client 0
09:50:36 SW1 BGP[30060]: SFL: [bgpd] LicenseCheck: ACCESS is active
09:50:36 SW1 BGP[30060]: SFL: [bgpd] LicenseCheck: returns Success.
09:50:36 SW1 BGP[30060]: SFL: [bgpd] LicenseCheck: BGP-FULL is active
09:50:36 SW1 BGP[30060]: SFL: [bgpd] LicenseCheck: returns Success.
09:50:36 SW1 BGP[30060]: [SFL] route limit of 1000000 applied
09:50:36 SW1 BGP[30060]: BGPd 7.8.2 starting: vty@2605, bgp@179
09:50:38 SW1 BGP[30060]: 2001:db8:a:10::2-Outgoing [FSM] Ignoring event 1 in state Idle -
configuration not complete
09:50:38 SW1 BGP[30060]: 2001:db8:a:10::2-Outgoing [FSM] Ignoring event 2 in state Idle -
configuration not complete
09:50:38 SW1 BGP[30060]: 2001:db8:b::1-Outgoing [FSM] Ignoring event 1 in state Idle -
configuration not complete
09:50:38 SW1 BGP[30060]: 2001:db8:b::1-Outgoing [FSM] Ignoring event 35 in state Idle -
configuration not complete
09:50:38 SW1 BGP[30060]: 2001:db8:b::1-Outgoing [FSM] Ignoring event 35 in state Idle -
configuration not complete
09:50:38 SW1 BGP[30060]: 2001:db8:b::1-Outgoing [FSM] Ignoring event 35 in state Idle -
configuration not complete
09:50:38 SW1 BGP[30060]: 2001:db8:a:10::2-Outgoing [FSM] Ignoring event 35 in state Idle -
configuration not complete
09:50:38 SW1 BGP[30060]: 2001:db8:a:10::2-Outgoing [FSM] Ignoring event 35 in state Idle -
configuration not complete
09:50:38 SW1 BGP[30060]: 2001:db8:b::1-Outgoing [FSM] Ignoring event 35 in state Idle -
configuration not complete
09:50:38 SW1 BGP[30060]: 2001:db8:b::1-Outgoing [FSM] Ignoring event 35 in state Idle -
configuration not complete
09:50:38 SW1 BGP[30060]: 2001:db8:b::1-Outgoing [FSM] Ignoring event 35 in state Idle -
configuration not complete
```

- Debug on SW1 ends here because BGP debug stops when daemon is restarted.

SW2's View:

- SW2 sends BGP Updates containing its prefixes once the connection is back up.
- SW2 sends End-of-RIB marker.

```
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Msg-Hdr: Type 2
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Update: AFI/SAFI (2/1) Tot-attr-len 7
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Update: Msg #58 Size 30
```

- SW2 receives updates from SW1, followed by an End-of-RIB marker.

```
09:50:36 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [NETWORK] FD=14, Sock Status: 107-Transport
endpoint is not connected
09:50:36 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] State: Established Event: 18
09:50:36 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] State Change: Established(6)->Idle(1)
09:50:36 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [GRST] Action Established: Restart Timer(90)
started, Event 18
09:50:36 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] Auto-Start Timer Expiry
09:50:36 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] State: Idle Event: 3
09:50:36 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] State Change: Idle(1)->Connect(2)
09:50:36 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [NETWORK] FD=14, Sock Status: 111-Connection
refused
09:50:36 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] State: Connect Event: 18
09:50:36 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] State Change: Connect(2)->Active(3)
09:50:38 SW2 BGP[1854]: [NETWORK] Accept Thread: Incoming conn from host 2001:db8:a::1 (FD=14)
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] State: Active Event: 14
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] InConnReq: Accepting...
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [NETWORK] FD=14, Sock Status: 0-Success
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] State: Active Event: 17
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Msg-Hdr: Type 1
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Open: Ver 4 MyAS 200 Holdtime 90
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Open: Msg-Size 71
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] State Change: Active(3)->OpenSent(4)
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Msg-Hdr: type 1, length 71
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Msg-Hdr: Requesting immediate Read (52)
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Open: Optional param len 42
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Open Opt: Option Type 2, Option Len 6
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Open Cap: Cap Code 1, Cap Len 4
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Open Opt: Option Type 2, Option Len 6
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Open Cap: Cap Code 1, Cap Len 4
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Open Opt: Option Type 2, Option Len 2
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Open Cap: Cap Code 128, Cap Len 0
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Open Cap: RR Cap(old) for all address-
families
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Open Opt: Option Type 2, Option Len 2
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Open Cap: Cap Code 2, Cap Len 0
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Open Cap: RR Cap(new) for all address-
families
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Open Opt: Option Type 2, Option Len 6
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Open Cap: Cap Code 65, Cap Len 4
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Open Opt: Option Type 2, Option Len 8
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Open Cap: Cap Code 64, Cap Len 6
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Cap GR: Restart Flag On,
```

[Continued on next page...]

```
Restart Time 90
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Cap GR: AFI/SAFI 2/1 Fwd-state Flag 1,
action: Set
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] State: OpenSent Event: 19
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Msg-Hdr: Type 4
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Keepalive: 93 KAlive msg(s) sent
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] State Change: OpenSent(4)->OpenConfirm(5)
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Msg-Hdr: type 4, length 19
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Msg-Hdr: Requesting immediate Read (0)
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] KAlive: Received!
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] State: OpenConfirm Event: 26
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [GRST] Initial Announce: Stalepath Preserve
Timer(360) started
09:50:38 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] State Change: OpenConfirm(5)->Established(6)
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] AS-Origination Timer Expiry
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] Routeadv Timer Expiry)
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] State: Established Event: 33
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] State: Established Event: 34
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Msg-Hdr: Type 2
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Update MP Reach: Prefix 2001:db8:1::/64
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Update MP Reach: Prefix 2001:db8:b:20::/
64
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Update MP Reach: Prefix 2001:db8:b:10::/
64
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Update MP Reach: Prefix 2001:db8:b::1/128
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Update MP Reach: Prefix 2001:db8:4::/64
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Update MP Reach: Prefix 2001:db8:3::/64
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Update: AFI/SAFI (2/1) Tot-attr-len 100
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Update: Msg #56 Size 123
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Msg-Hdr: Type 2
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Update MP Reach: Prefix 2001:db8:b:30::/
64
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Update MP Reach: Prefix 2001:db8:b:40::/
64
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Update: AFI/SAFI (2/1) Tot-attr-len 63
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Update: Msg #57 Size 86
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Msg-Hdr: Type 2
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Update: AFI/SAFI (2/1) Tot-attr-len 7
09:50:39 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Update: Msg #58 Size 30
09:50:39 SW2 BGP[1854]: [RIB] Scanning BGP RIB...
09:50:39 SW2 BGP[1854]: NSM Message Header
09:50:39 SW2 BGP[1854]:  VR ID: 0
09:50:39 SW2 BGP[1854]:  VRF ID: 0
09:50:39 SW2 BGP[1854]:  Message type: IPv6 Route (32)
09:50:39 SW2 BGP[1854]:  Message length: 76
09:50:39 SW2 BGP[1854]:  Message ID: 0x00000061
09:50:39 SW2 BGP[1854]: NSM IPv6 route add
09:50:39 SW2 BGP[1854]:  Flags: 1
09:50:39 SW2 BGP[1854]:  Route: 2001:db8:a::1/128
09:50:39 SW2 BGP[1854]:  Type: 3
09:50:39 SW2 BGP[1854]:  Metric: 0
09:50:39 SW2 BGP[1854]:  Distance: 1
```

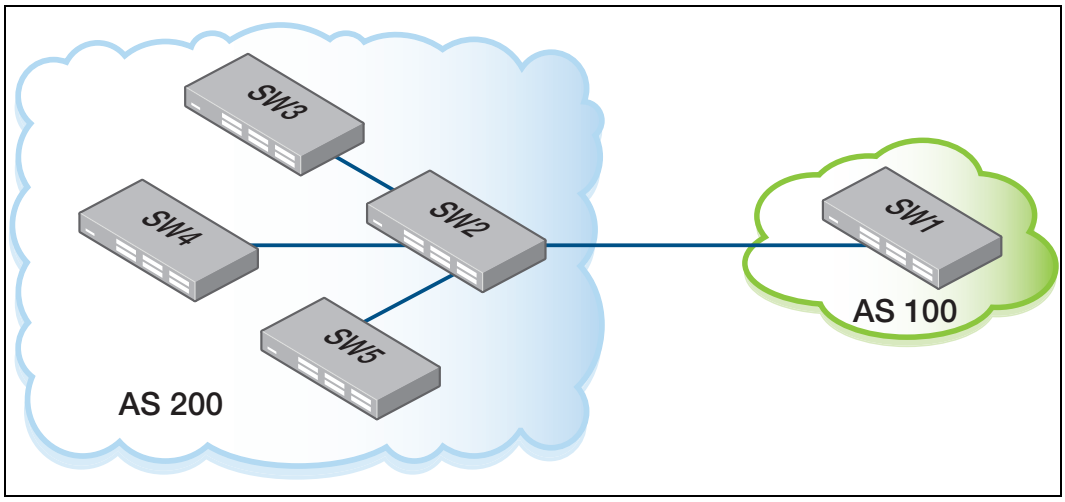
[Continued on next page...]

```
09:50:39 SW2 BGP[1854]:  Nexthop: fe80::200:cdff:fe37:f0c ifindex 400
09:50:39 SW2 BGP[1854]: [NSM] Verified NH 2001:db8:a::1 with NSM
09:50:39 SW2 BGP[1854]: [RIB] Scanning BGP Network Routes...
09:50:44 SW2 BGP[1854]: [NETWORK] Accept Thread: Incoming conn from host 2001:db8:c::1 (FD=15)
09:50:44 SW2 BGP[1854]: 2001:db8:c::1-Outgoing [FSM] State: Idle Event: 14
09:50:44 SW2 BGP[1854]: 2001:db8:c::1-Outgoing [FSM] InConnReq: Ignoring...
09:50:54 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] AS-Origination Timer Expiry
09:50:54 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] State: Established Event: 33
09:50:54 SW2 BGP[1854]: [RIB] Scanning BGP Network Routes...
09:51:03 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] Routeadv Timer Expiry)
09:51:03 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] State: Established Event: 34
09:51:04 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] Keep-alive-Timer Expiry
09:51:04 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] State: Established Event: 11
09:51:04 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Msg-Hdr: Type 4
09:51:04 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [ENCODE] Keepalive: 94 KAlive msg(s) sent
09:51:07 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Msg-Hdr: type 2, length 143
09:51:07 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Msg-Hdr: Requesting immediate Read (124)
09:51:07 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Update: Starting UPDATE decoding... Bytes
To Read (124), msg_size (124)
09:51:07 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] State: Established Event: 27
09:51:07 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [RIB] Update: Received Prefix 2001:db8:1::/64
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_withdraw(): No dampening reqd
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_withdraw(): Route State: NONE, ret=0
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_update(): Route State: NONE, ret=0
09:51:07 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [RIB] Update: Received Prefix ::/0
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_withdraw(): No dampening reqd
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_withdraw(): Route State: NONE, ret=0
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_update(): Route State: NONE, ret=0
09:51:07 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [RIB] Update: Received Prefix 2001:db8:a:70::/64
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_withdraw(): No dampening reqd
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_withdraw(): Route State: NONE, ret=0
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_update(): Route State: NONE, ret=0
09:51:07 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [RIB] Update: Received Prefix 2001:db8:a:60::/64
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_withdraw(): No dampening reqd
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_withdraw(): Route State: NONE, ret=0
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_update(): Route State: NONE, ret=0
09:51:07 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [RIB] Update: Received Prefix 2001:db8:a:50::/64
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_withdraw(): No dampening reqd
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_withdraw(): Route State: NONE, ret=0
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_update(): Route State: NONE, ret=0
09:51:07 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [RIB] Update: Received Prefix 2001:db8:a:40::/64
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_withdraw(): No dampening reqd
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_withdraw(): Route State: NONE, ret=0
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_update(): Route State: NONE, ret=0
09:51:07 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [RIB] Update: Received Prefix 2001:db8:a:30::/64
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_withdraw(): No dampening reqd
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_withdraw(): Route State: NONE, ret=0
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_update(): Route State: NONE, ret=0
09:51:07 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [RIB] Update: Received Prefix 2001:db8:a:20::/64
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_withdraw(): No dampening reqd
09:51:07 SW2 BGP[1854]: [DAMP] bgp_rfd_rt_withdraw(): Route State: NONE, ret=0
09:51:07 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Msg-Hdr: type 2, length 30
09:51:07 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Msg-Hdr: Requesting immediate Read (11)
09:51:07 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [DECODE] Update: Starting UPDATE decoding... Bytes
To Read (11), msg_size (11)
09:51:07 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] State: Established Event: 27
09:51:07 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] Update: AFI-SAFI:2-1 End-Of-RibMarker
Received
09:51:07 SW2 BGP[1854]: 2001:db8:a::1-Outgoing [FSM] Process End-of-RIB: Received for afi/safi: 2/
```

Using peer groups to simplify administration and conserve processing and memory

Scenario

Example showing implementations of iBGP and eBGP peer groups.



Use

- Optimize CPU and memory resources.
- Make updating more efficient to neighbors which have identical update policies.

When to use peer groups

- When there are many iBGP or eBGP peers which share the same update policies.
- Alternative method to defining the same policies per peer separately, peers can be grouped into a peer group.
- A single policy can be configured once for the peer group, and will apply to all peers in the group.
- A BGP update needs to be generated only once for the entire peer group, instead of once per neighbor. (RIB requires only one scan)

Implementation example

- A device is to be the Next Hop for iBGP peers (using the next-hop-self command).
- This can be applied to the peer group, instead of per neighbor.
- This simplifies configuration, and the device will be the Next Hop for all iBGP peers in the peer group.

Limitations

eBGP and iBGP peers cannot be grouped into the same peer group.

```
SW2(config-router-af)#neighbor 2001:db8:10::1 peer-group IBGP-PGROUP
% Peer with AS 100 cannot be in this peer-group, members must be all internal
or all external
```

Objective

- Setup and configure a BGP peer group for IPv6 iBGP peers.
- SW2 is to be a route reflector and the Next Hop for all iBGP peers.

Example: Configuring BGP without using peer groups

- Configuration is repetitive.
- Policies and configuration are applied sequentially, using more CPU and memory resources.

```
SW2(config)#router bgp 200
SW2(config-router)#neighbor 2001:db8:10::1 remote-as 100
SW2(config-router)#neighbor 2001:db8:20::2 remote-as 200
SW2(config-router)#neighbor 2001:db8:30::2 remote-as 200
SW2(config-router)#neighbor 2001:db8:40::2 remote-as 200
SW2(config-router)#address-family ipv6
SW2(config-router-af)#neighbor 2001:db8:10::1 activate
SW2(config-router-af)#neighbor 2001:db8:20::2 activate
SW2(config-router-af)#neighbor 2001:db8:30::2 activate
SW2(config-router-af)#neighbor 2001:db8:40::2 activate
SW2(config-router-af)#neighbor 2001:db8:20::2 next-hop-self
SW2(config-router-af)#neighbor 2001:db8:30::2 next-hop-self
SW2(config-router-af)#neighbor 2001:db8:40::2 next-hop-self
SW2(config-router-af)#neighbor 2001:db8:20::2 route-reflector-client
SW2(config-router-af)#neighbor 2001:db8:30::2 route-reflector-client
SW2(config-router-af)#neighbor 2001:db8:40::2 route-reflector-client
```

Creating a BGP peer group

1. Create a peer group for the iBGP peers.

```
SW2(config)#router bgp 200
SW2(config-router)#neighbor IBGP-PGROUP peer-group
SW2(config-router)#neighbor IBGP-PGROUP remote-as 200
```

2. Activate the peer group for the IPv6 address family.

```
SW2(config-router)#address-family ipv6
SW2(config-router-af)#neighbor IBGP-PGROUP activate
```

3. Assign peers to the peer group.

```
SW2(config-router-af)#neighbor 2001:db8:20::2 peer-group IBGP-PGROUP
SW2(config-router-af)#neighbor 2001:db8:30::2 peer-group IBGP-PGROUP
SW2(config-router-af)#neighbor 2001:db8:40::2 peer-group IBGP-PGROUP
```

4. Assign group policies for the peer group.

```
SW2(config-router-af)#neighbor IBGP-PGROUP next-hop-self
SW2(config-router-af)#neighbor IBGP-PGROUP route-reflector-client
```

Post peer-group configuration

Notice that instead of the next-hop-self and route-reflector policies applied to each peer, the policies are instead applied to the peer group.

```
SW2#show run bgp
!
router bgp 200
bgp router-id 2.2.2.2
no bgp default ipv4-unicast
neighbor IBGP-PGROUP peer-group
neighbor IBGP-PGROUP remote-as 200
neighbor 2001:db8:10::1 remote-as 100
!
address-family ipv6
neighbor IBGP-PGROUP activate
neighbor IBGP-PGROUP next-hop-self
neighbor 2001:db8:10::1 activate
neighbor 2001:db8:20::2 peer-group IBGP-PGROUP
neighbor 2001:db8:30::2 peer-group IBGP-PGROUP
neighbor 2001:db8:40::2 peer-group IBGP-PGROUP
exit-address-family
!
```

Post peer group configuration

Notice that instead of the next-hop-self and route-reflector policies applied to each peer, the policies are instead applied to the peer group.

```
SW2#show run bgp
!
router bgp 200
bgp router-id 2.2.2.2
neighbor IBGP-PGROUP peer-group
neighbor IBGP-PGROUP remote-as 200
neighbor 2001:db8:10::1 remote-as 100
!
address-family ipv6
neighbor IBGP-PGROUP activate
neighbor IBGP-PGROUP next-hop-self
neighbor 2001:db8:10::1 activate
neighbor 2001:db8:20::2 peer-group IBGP-PGROUP
neighbor 2001:db8:30::2 peer-group IBGP-PGROUP
neighbor 2001:db8:40::2 peer-group IBGP-PGROUP
exit-address-family
```

Verify

Neighbor session information shows the peer is a member of the iBGP-PGROUP.

```
SW2#show bgp ipv6 neighbors 2001:db8:20::2
BGP neighbor is 2001:db8:20::2, remote AS 200, local AS 200, internal link
 Member of peer-group IBGP-PGROUP for session parameters
  BGP version 4, remote router ID 3.3.3.3
  BGP state = Established, up for 00:01:59
  Last read 00:01:59, hold time is 90, keepalive interval is 30 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    4-Octet ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
 Received 11 messages, 0 notifications, 0 in queue
 Sent 12 messages, 1 notifications, 0 in queue
 Route refresh request: received 0, sent 0
 Minimum time between advertisement runs is 5 seconds
For address family: IPv6 Unicast
 BGP table version 2, neighbor version 2
 Index 2, Offset 0, Mask 0x4
 IBGP-PGROUP peer-group member
 Route-Reflector Client
 NEXT_HOP is always this router
 Community attribute sent to this neighbor (both)
 0 accepted prefixes
 1 announced prefixes

 Connections established 2; dropped 1
Local host: 2001:db8:20::1, Local port: 179
Foreign host: 2001:db8:20::2, Foreign port: 44447
Nexthop: 2.2.2.2
Nexthop global: 2001:db8:20::1
Nexthop local: fe80::eecd:6dff:fe20:c26b
BGP connection: shared network
Last Reset: 00:01:59, due to BGP Notification sent
Notification Error Message: (Cease/Other Configuration Change.)
```

Additional

Remove a peer from the peer group.

```
SW2(config)#router bgp 200
SW2(config-router)#address-family ipv6
SW2(config-router-af)#no neighbor 2001:db8:20::2 peer-group IBGP-PGROUP
```

Clear the peer group sessions.

- Specify the peer group for which the sessions should be cleared.
- Standard session clearing options are available.

```
SW2#clear bgp ipv6 peer-group ?
WORD BGP peer-group name

SW2#clear bgp ipv6 peer-group IBGP-PGROUP ?
in Soft reconfig inbound update
out Soft reconfig outbound update
soft Soft reconfig
```

Additional Information

For more in-depth details about the commands used in these examples, or the outputs from validation commands, see Chapter 3 of the BGP and BGP4+ Software Reference: BGP & IPv6 (Border Gateway Protocol IPv4 & IPv6) Supplement for x-Series Switches, AlliedWare Plus™ Operating System Version 5.4.3-2.56.

To find this document, go to alliedtelesis.com and in the Documentation area, enter the word **Supplement** in the search box.

How to Note

The following Allied Telesis How to Note may also be of interest:

- How to Use Route Maps and Other Filters to Filter and Alter BGP and OSPF Routes

To find this document, go to alliedtelesis.com and in the Documentation area, enter the words **Route Maps** in the search box.

Additional Allied Telesis Technical Guides

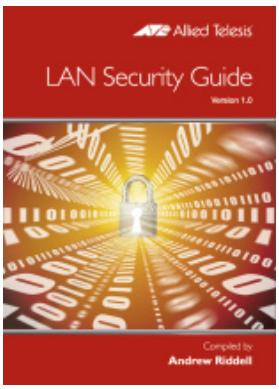
Other technical guides available from Allied Telesis include:

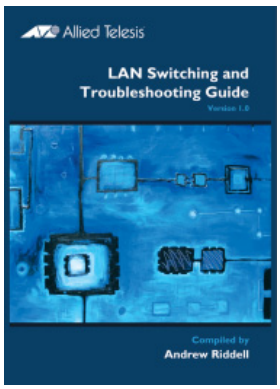
LAN Security Guide

This guide provides a mixture of practical advice on secure configuration of network devices, and explanations of the protocols and technologies that are employed in LAN security.

LAN Switching and Troubleshooting Guide

This guide explains the basics of LAN switching and provides a high-level view of the fundamental components and key protocols used to help smooth data flow.





BGP4 Protocol information

For more information on the BGP4 protocol, refer to: http://www.ietf.org/rfc/rfc4271.txt