



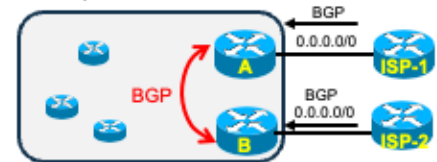
Enterprise Intra-AS BGP Peering



Reasons For BGP In The Enterprise

- + People typically implement BGP on two-or-more routers within an Enterprise (government network, campus, etc) for the following reasons:

- + Multihoming to ISPs...propagate default routes



- + Transit areas to allow connectivity to Partner/Acquisition networks running different IGPs



- + Scalability for very large networks



BGP Enterprise Peering Considerations

- + When multiple routers within an ASN will run BGP you must choose how they will peer with each other;
 - + eBGP between all routers
 - + iBGP between all routers
- + There are pros and cons with each
- + Key things to consider;
 - + Will an IGP be present for next-hop/peer reachability?
 - + iBGP route propagation rules
 - + ASN numbering implementation for eBGP networks



iBGP Between Intra-AS Peers

- + Why choose iBGP between intra-AS peers?
 - + Allows use of **Local-Preference** for best path manipulation
 - + Defers to paths learned via IGP's preventing non-optimal path selection when both IGP and BGP are used simultaneously
 - + Allow one to maintain public ASN usage without the **complexities of introducing private ASNs**
 - + Fewer BGP "neighbor" statements required when implementing Route-Reflectors than eBGP-Only Enterprises.



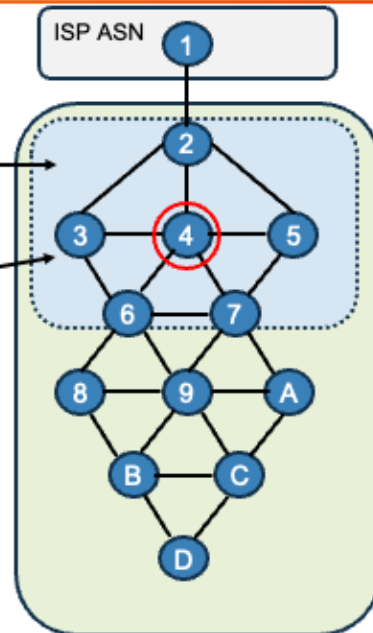
eBGP Between Intra-AS Peers

- + Why choose eBGP between intra-AS peers?
- + eBGP peering alleviates many of the extra mechanisms and features required in an iBGP network
 - + No extra Route-Reflector configuration required
 - + No need for static routes or IGP
 - + BGP next-hop is automatically reachable between eBGP peers
- + Shortest path automatically selected via AS-Path length
- + **NOTE:** Assigning a unique ASN to each router in a large enterprise will result in longer AS path lengths per prefix (depending on end-to-end hop count)
 - + Per RFC-1771 the longest allowable AS-Path length one can receive is **255** ASNs



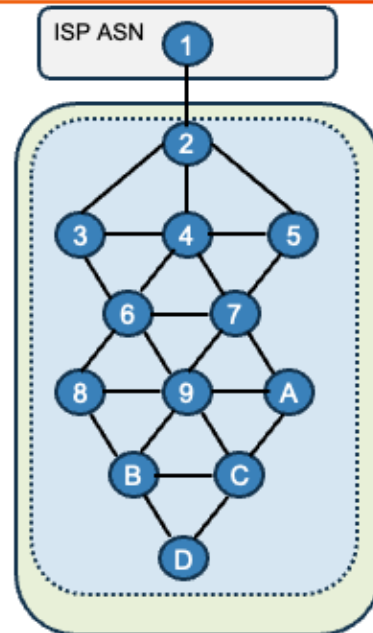
Contrasting Peering Quantities

- + Assumed goal: To achieve maximum path redundancy and load-balancing
- + eBGP between all routers
 - + **21** "neighbor" statements
 - + No IGP or static routes required
- + iBGP with R4 as Route Reflector
 - + 1-eBGP "neighbor" statement on R2 (towards R1)
 - + 10-iBGP "neighbor" statements
 - + **11** "neighbor" statements total
 - + IGP or static routes required for next-hop reachability



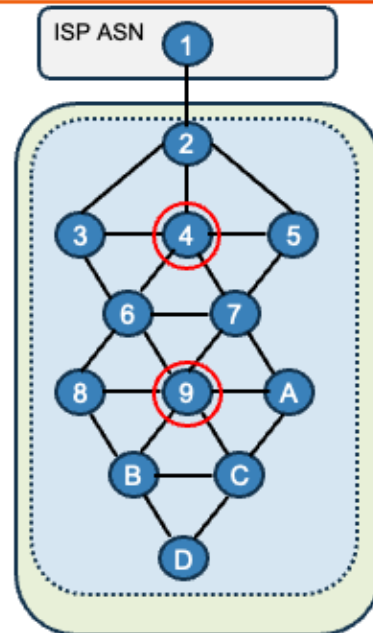
Contrasting “Neighbor” Quantities

- + Assumed goal: To achieve maximum path redundancy and load-balancing
- + eBGP with directly connected peers
 - + No static routes or IGP required
 - + **47** “neighbor” statements



Contrasting "Neighbor" Quantities

- + eBGP with directly connected peers
 - + No static routes
 - + **47 "neighbor" statements total**
- + iBGP with R4 and R9 as Route Reflectors
 - + 1-eBGP "neighbor" statement on R2 (towards R1)
 - + 21-iBGP "neighbor" statements
 - + **22 "neighbor" statements total**
 - + IGP required for next-hop reachability



Enterprise Usage of Private ASNs

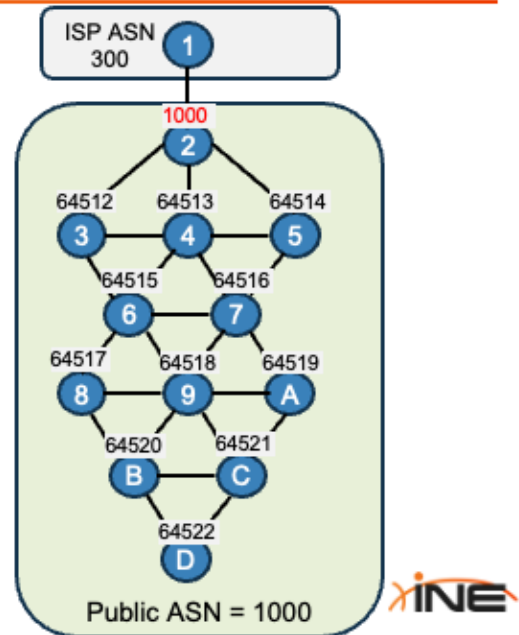
- + A common design when utilizing eBGP within the Enterprise is to allocate Private ASNs to each router.
 - + Adhering to the 2-byte Private range allows eBGP usage on up to 1,022 BGP-speaking devices
 - + Utilizing “allowas-in” doubles that number if there are no loops
- + Private ASN values must be removed from the BGP path before updates can be sent to Service Providers



- Remember the Private range for BGP ASNs is 64512 – 65534 (with 64511 and 65535 reserved for special use)

eBGP Enterprise Design

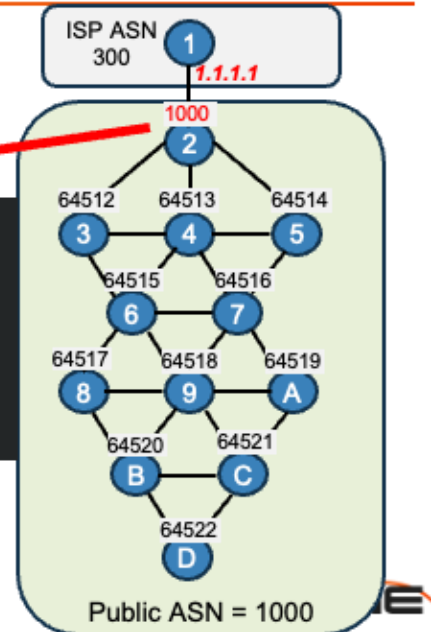
- + In this design router R2 must have the ability to remove all Private ASNs from BGP paths prior to sending BGP updates to the ISP
- + R2 must also advertise itself with a public ASN



BGP Remove-Private-AS

```
R2(config)#router bgp 1000
R2(config-router)#neighbor 1.1.1.1 remote-as 300
R2(config-router)#neighbor 1.1.1.1 remove-private-as ?
  all    Remove all private AS numbers
  <cr>  <cr>

R2(config-router)#neighbor 1.1.1.1 remove-private-as all ?
  replace-as Replace all private AS numbers with local AS
  <cr>      <cr>
```



- Without the “all” keyword, Private ASNs will NOT be removed from any path that R2 received containing a mixture of Private and Public ASNs, however in this topology we wouldn't expect R2 to receive any paths like that.
- This command ONLY dynamically recognizes Private ASNs from the 2-byte allocation. It does NOT recognize Private 4-byte ASNs in the range of 4200000000 to 4294967294.

For More Info

- + A great RFC to learn how to leverage BGP within the Enterprise and Data Center: **RFC 7938**

Internet Engineering Task Force (IETF)
Request for Comments: 7938
Category: Informational
ISSN: 2070-1721

P. Lapukhov
Facebook
A. Premji
Arista Networks
J. Mitchell, Ed.
August 2016

Use of BGP for Routing in Large-Scale Data Centers

Abstract

Some network operators build and operate data centers that support over one hundred thousand servers. In this document, such data centers are referred to as "large-scale" to differentiate them from smaller infrastructures. Environments of this scale have a unique set of network requirements with an emphasis on operational simplicity and network stability. This document summarizes operational experience in designing and operating large-scale data centers using BGP as the only routing protocol. The intent is to report on a proven and stable routing design that could be leveraged by others in the industry.







BGP Communities



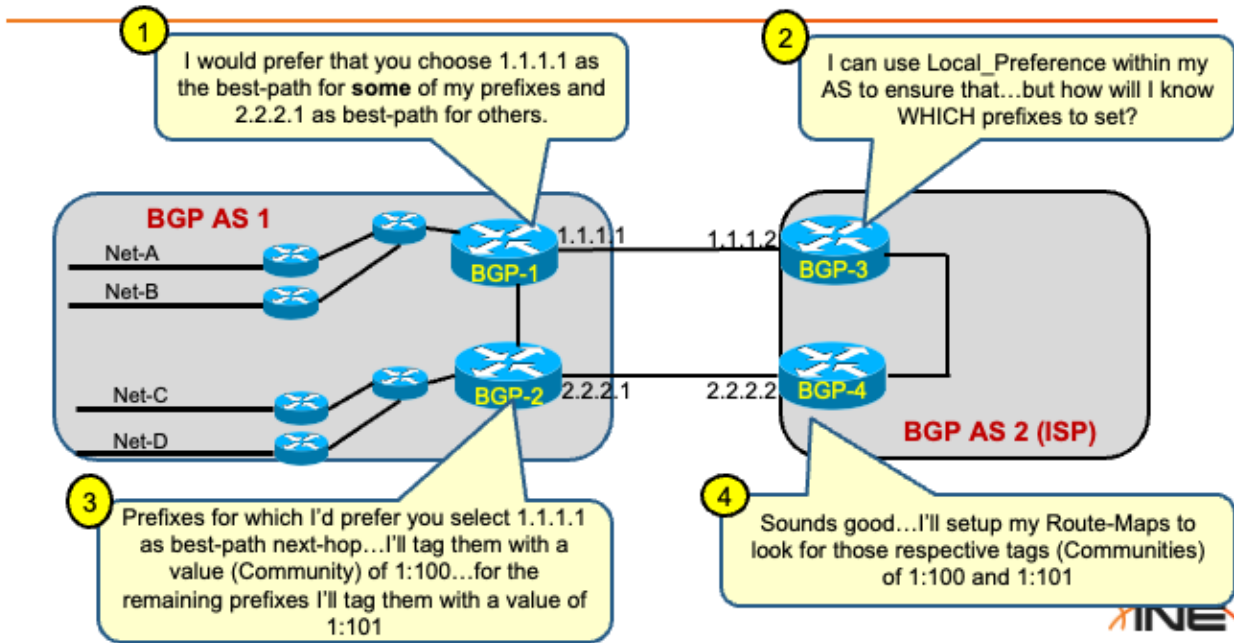
The BGP Community Path Attribute

- + Descriptive number/value that is applied as a “tag” to a route.
 - + Standard Community (RFC 1997) = 32-bits/4-bytes long
 - + Extended Community (RFC 4360) = 64-bits/8-bytes long
 - + Typical format **AS:value**
- + Used to group destinations and apply a common policy
- + Each prefix can belong to multiple communities
- + Some values are “Well-Known” and are understood to have special pre-defined meanings.



- Not part of the BGP bestpath algorithm...more like a route-tag that can be matched upon.
- -
- Considered, “Optional-Transitive”
- -
- Set-community <1-4294967295> or AA:NN where “nn” can be up to 4-digits.
- -
- Set extcommunity : Does NOT allow you to set your own value. Extended communities in IOS are for pre-defined usage only like SOO or RT.

Use-Case for BGP Communities



Well-Known BGP Communities

- + **internet** = (value = 0x0) all routes are members of this community
- + **Graceful shutdown** = (value = 65535:0) Peering is about to cease.
- + **no-export** = (value = 0xFFFFFFF01) do not advertise to eBGP peers
- + **no-advertise** = (value = 0xFFFFFFF02) do not advertise to any peer
- + **local-AS** = (value = 0xFFFFFFF03) do not advertise outside local AS (used with confederations)



- The “Graceful Shutdown” community is sent within a BGP Update packet after the command, “neighbor x.x.x.x shutdown graceful <time> community <value>” is configured. Typically used when planned maintenance will shutdown a link. Receiving routers can match on this community and filter routes so replacement routes can immediately take affect before the peering is lost.
- No-Export does NOT prevent sending a prefix between Confederation-External peers...hence the need for “Local-AS”.

Configuring Communities

```
R7(config)#route-map Test permit 10
R7(config-route-map)#set community ?
<1-4294967295> community number
aa:nn          community number in aa:nn format
gshut          Graceful Shutdown (well-known community)
internet       Internet (well-known community)
local-AS       Do not send outside local AS (well-known community)
no-advertise   Do not advertise to any peer (well-known community)
no-export      Do not export to next AS (well-known community)
none           No community attribute
```



Transmitting BGP Communities

```
router bgp 333
neighbor x.x.x.x send-community
neighbor x.x.x.x route-map Community out
!
access-list 1 permit 100.100.0.0 0.0.255.255
!
route-map Community permit 10
match ip address 1
set community 100100
!
route-map Community permit 20
```

Must explicitly give BGP permission to send Communities to BGP neighbors.

In this example, a BGP Community is added to outbound routes as stated in a Route map

This is just a generic BGP Community number (a tag) that has no implicit meaning to the sender...only the receiver.



- KEY POINT: Communities are NOT propagated between neighbors by default!!
- -
- Received a route with an existing Community that you want to strip out? "set community none" in Route-Map.

Matching on BGP Communities

```
router bgp 1112
neighbor 192.168.1.17 route-map INE in
!
ip community-list 1 permit 100100
!
route-map INE permit 10
match community 1
set weight 300
!
route-map INE permit 20
!
```

Inbound BGP Communities are matched using this command (Standard Community)

In this example, any route received that is tagged with BGP Standard Community value 100100 is given a BGP Weight of 300.



- Community-Lists support regular-expressions if needed.

Verifying Communities

```
R7(config)#route-map Attribs permit 10
R7(config-route-map)#set community 7:200
R7(config-route-map)#exit
R7(config)#router bgp 7
R7(config-router)#neighbor 22.7.22.2 send-community both
```

On sending router...

```
R2#show ip bgp 77.77.77.0/24
```

On receiving router...

```
 7 6688 123 4458 1001 3002 78009 65
 22.7.22.7 from *22.7.22.7 (22.7.22.7)
   Origin IGP, metric 0, localpref 100, valid, external
   Community: 458952
   rx pathid: 0, tx pathid: 0
```



BGP New-Format

- + BGP communities are normally displayed as a string of digits in the output of BGP “show” commands

```
R2#show ip bgp 7.1.7.0/24
BGP routing table entry for 7.1.7.0/24, version 31
 7
 7.7.7.7 from 7.7.7.7 (166.66.66.5)
  Origin IGP, metric 0, localpref 100, valid, external, best
  Community: 458759
 rx pathid: 0, tx pathid: 0x0
```

- + Use the Global Configuration command, “*ip bgp new-format*” if you wish to view BGP Communities as x:y in command output.

```
R2(config)#ip bgp new-format
7.7.7.7 from 7.7.7.7 (166.66.66.5)
  Origin IGP, metric 0, localpref 100, valid, external, best
  Community: 7:7
 rx pathid: 0, tx pathid: 0x0
```







BGP Prefix Filtering Techniques



BGP Prefix-Based Filtering

- + Prefix filtering requires utilizing a tool that will match on the prefixes that interest you
- + Classification Tools Available
 - + Access-Lists
 - ✓ Standard – For matching on prefix only
 - ✓ Extended – For matching on prefix and subnet masks
 - + Prefix-Lists
- + Where to apply ACLs/Prefix-Lists?
 - + Directly against BGP “neighbor” statements (Prefix-Lists)
 - + Distribute-Lists (ACLs & Prefix-Lists)
 - + Route-Maps (ACLs or Prefix-Lists)



- Distribute-Lists can only reference Prefix-Lists if applied within BGP “Address-Family” modes.

A Review of Distribute-Lists

- + A popular tool for applying prefix-based filtering is the Distribute-List
- + Distribute-Lists must be associated with directionality.
 - + Inbound: Permit or deny prefixes received in BGP updates
 - + Output: Permit or deny prefixes sent to BGP peers
- + Distribute-Lists must reference a classification tool
 - + Access-Lists
 - + Prefix-Lists
 - + Route-Maps
- + If a prefix is “permitted” by the route classification tool, it will be “distributed” (inbound or outbound) by the Distribute-List



Filtering with Standard ACLs

Standard ACLs can only match against the prefix...not the subnet mask associated with that prefix.

» Example 1:

```
router bgp 10
neighbor x.x.x.x distribute-list 1 in
!
access-list 1 deny 160.10.0.0 0.0.255.255
access-list 1 permit any
```

Objective: Deny any advertised prefix that starts with the first 16-bits matching 160.10.0.0 (subnet mask irrelevant).

» Example 2:

```
router bgp 10
neighbor x.x.x.x route-map Standard-ACL in
!
route-map Standard-ACL permit 10
match ip address 1
!
access-list 1 deny 160.10.0.0 0.0.255.255
access-list 1 permit any
```



Filtering with Extended ACLs

» Use 'extended access-lists' in BGP to match on prefix and subnet mask(s)

» Example:

```
router bgp 10
neighbor x.x.x.x distribute-list 101 in
!
access-list 101 deny ip 160.0.0.0 0.255.255.255 255.255.240.0 0.0.15.255
access-list 101 permit ip any any
```

Goal: Deny any advertised prefix beginning with 160.x.x.x that has a subnet mask of at least 20-bits.

Prefix advertised by neighbor

We only care about first 8-bits of this prefix.

Mask associated with that prefix

Any mask that is at least a /20



- Distribute-Lists can also be applied directly in address-family mode without applying against a neighbor. In this way ALL neighbors are affected at once.
- -
- Within Address-family mode ONLY, distribute-Lists can also reference prefix-lists (but NOT route-maps).

Filtering with Prefix-Lists

» Within BGP, Prefix-Lists can be utilized for prefix-based filtering:

- As a classification tool within a Route-Map sequence
- Directly against a BGP “neighbor” statement

» Example:

```
ip prefix-list Test deny 150.0.0.0/8 ge 20 le 24
ip prefix-list Test permit 0.0.0.0/0 le 32
!
router bgp 10
neighbor x.x.x.x prefix-list Test in
```

Deny any prefix where the first 8-bits match 150.x.x.x and the subnet mask is from /20 through /24.



- Once you configure BGP in “address-family” mode, distribute-lists can no longer be applied directly against BGP “neighbor” statements.

BGP Distribute-List Caveats

- + As long as BGP has not been configured with any address-families, Distribute-Lists can be applied directly against BGP “neighbor” statements.

```
R7(config)#router bgp 7
R7(config-router)#neighbor 2.2.2.2 distribute-list ?
<1-199>      IP access list number
<1300-2699>  IP access list number (expanded range)
WORD         IP Access-list name
```

- + Once Address-Family configuration has been entered, Distribute-Lists against peers **must be applied within the Address-Family**:

```
R7(config-router)#address-family ipv4
R7(config-router-af)#distribute-list ?
<1-199>      IP access list number
<1300-2699>  IP expanded access list number
WORD         Access-list name
gateway      Filtering incoming updates based on gateway
prefix       Filter prefixes in routing updates

R7(config-router-af)#exit
R7(config-router)#neighbor 2.2.2.2 distribute-list ?
% Unrecognized command
```



- Here we can see that after an Address-Family for IPv4 has been configured, beneath the global BGP process the “distribute-list” command is no longer available to apply against BGP neighbors. If that is what you wish to do, you must enter the appropriate Address-Family to do it.


Filtering with Route-Maps & Prefix Lists

- » Route-Maps that reference Prefix-Lists can be used for prefix-based filtering.

```
ip prefix-list Test permit 150.0.0.0/8 ge 20 le 24
!
Route-map Filter deny 10
  match ip address prefix Test
Route-map Filter permit 20
!
router bgp 10
neighbor x.x.x.x route-map Filter in
```







BGP Prefix Summarization (Auto-Summary)



BGP Prefix Summarization

- + In large enterprise networks with huge data centers, tens of thousands of prefixes may be generated
- + To shrink the size of BGP tables, it is often desirable to summarize contiguous prefixes.
- + Prefixes must first be injected into the local BGP table before they can be summarized



Aggregation - Methods

+ Two methods to aggregate prefixes:

- + "Aggregate-address" command
- + "Auto-summary" command

+ Auto-Summary

- + Useful when aggregating many IGP routes into a single, classful prefix
- + Works along with BGP "network" command without any mask.
- + Not possible to manipulate/define mask of summarized route.
- + Not used to aggregate learned-BGP prefixes.



The “Auto-Summary” command

- + Creates summarized prefix advertisement using classful masks
- + Works with Redistribution and the BGP “network” command
- + Does not affect BGP “network” statements that contain “mask” keywords
- + When **auto-summary** is configured, the logic of the network statement changes only if the **mask** parameter is omitted.
 - + With no **mask** parameter configured, and with **auto-summary** configured, the router adds a route for that classful network to the BGP table.
 - + Suppresses subnet routes



- If auto-summary is configured, and we use the mask keyword with a “network” statement, nothing happens. The network is advertised as normal and not summarized.
- It's when we DON'T use the mask keyword that the behavior changes
- If we don't use that keyword, and we configure auto-summary, we'll generate a classful route instead of the network and associated subnet mask we configured.

Example of “Auto-Summary”

```
R1#show ip route | i (200.200.*\./2.*)
      200.200.200.0/24 is variably subnetted, 8 subnets, 2 masks
C       200.200.200.0/26 is directly connected, Loopback3
C       200.200.200.64/26 is directly connected, Loopback4
C       200.200.200.128/26 is directly connected, Loopback5
C       200.200.200.192/26 is directly connected, Loopback6
```

```
router bgp 1
  bgp log-neighbor-changes
  network 200.200.200.0
  neighbor 1.2.1.2 remote-as 2
  auto-summary
```

```
R1#show ip bgp
BGP table version is 22, local router ID is 111.111.111.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
               x best-external, a additional-path, c RIB-compressed,
               t secondary path,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

   Network          Next Hop           Metric LocPrf Weight Path
* > 200.200.200.0    0.0.0.0              0         32768 i
R1#
```



- Here we can see that this router has routes to all possible subnets of the 200.200.200.x Class-C network.
- Rather than advertising the four subnets, we simply use the combination of a single “network” command along with “auto-summary” to advertise the entire Class-C network of 200.200.200.0/24





BGP Prefix Summarization (Aggregate-Address)



Aggregation-specific Path Attributes

+ When aggregating routes, a BGP speaker may choose to add one-or-more Optional-Discretionary Path Attributes:

+ **Aggregator Attribute**: (Optional-Transitive) AS number and Router-ID of the router that aggregated the route.

```
De-3#sho ip bgp 192.168.1.0/24
BGP routing table entry for 192.168.1.0/24, version 47
Paths: (3 available, best #1, table Default-IP-Routing-Table)
Flag: 0x820
Advertised to update-groups:
  1
(65501) (aggregated by 333 33.33.3.3)
```

+ **Atomic-Aggregate**: (Well-Known Discretionary) When AS-Set information is NOT included in the path, this attribute is inserted to indicate that this route is an aggregated prefix and all specific subset information has been lost.

```
(65502 65501), (aggregated by 333 33.33.3.3)
33.33.3.3 (metric 156160) from 33.33.11.11 (33.33.11.11)
Origin IGP, metric 0, localpref 100, valid, confed-internal, atomic-aggregate
```



- Atomic Aggregate is also Transitive.

BGP Aggregate-Address Command

+ Can be done in BGP process or Address-Family modes.

+ Without any keywords:

+ Creates/advertises aggregated prefix.

+ Does not advertise any AS-SET information.

+ Does insert Atomic-Aggregate PA.

+ Also allows advertisement of subset routes.

```
Rtr-3#sh run | s bgp
router bgp 65501
  bgp log-neighbor-changes
  bgp confederation identifier 333
  bgp confederation peers 65503
  aggregate-address 192.168.1.0 255.255.255.0
  neighbor 33.33.1.1 remote-as 65501
  neighbor 33.33.1.1 update-source Loopback0
```

```
Sw-3#sho ip bgp
BGP table version is 61, local router ID is 33.33.33.33
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network        Next Hop        Metric LocPrf Weight Path
*> 1.2.1.0/24      1.2.1.254         0      100      0 (65501) 222 7 99 2033 ?
*> 1.2.2.0/24      1.2.1.254         0      100      0 (65501) 222 7 99 2033 ?
*> 192.168.1.0     33.33.33.3         0      100      0 (65501) i
*> 192.168.1.4/30  1.2.1.254         0      100      0 (65501) 222 7 99 2033 i
*> 192.168.1.8/30  1.2.1.254         0      100      0 (65501) 222 500 23 i
*> 192.168.1.12/30 1.2.1.254         0      100      0 (65501) 222 500 23 ?
*> 192.168.1.16/28 1.2.1.254         0      100      0 (65501) 222 7 99 2033 ?
```



- All forms of aggregation using the “aggregate-address” command always cause the “Aggregator” PA to be inserted.
- Atomic Aggregate PA is ONLY inserted when AS-Set is NOT included as part of the path.
- -
- BGP Communities that were present in individual prefixes are included (compiled) into the aggregated route IF:
 - -----The “neighbor x.x.x.x send-community” command is configured.
 - -----The “as-set” or “as-confed-set” keywords are used.

Aggregate-Address – (Summary Only)

+Aggregate-address x.x.x.x <mask> *summary-only*

- +Transmits just the aggregated route...suppressing all subset routes.
- +Does NOT include AS-SET information
- +DOES include Atomic-Aggregate PA

```
router bgp 65501
  bgp log-neighbor-changes
  bgp confederation identifier 333
  bgp confederation peers 65503
  aggregate-address 192.168.1.0 255.255.255.0 summary-only
```

```
Sw-3#sho ip bgp
BGP table version is 65, local router ID is 33.33.33.33
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network        Next Hop           Metric LocPrf Weight Path
r> 1.2.1.0/24      1.2.1.254             0      100      0 (65501) 222 7 99 2033 ?
s> 1.2.2.0/24      1.2.1.254             0      100      0 (65501) 222 7 99 2033 ?
*> 192.168.1.0     33.33.33.3             0      100      0 (65501) i
Sw-3#
```

Aggregate-Address – (AS-SET)

+Aggregate-address x.x.x.x <mask> as-set

- +Includes AS-SET information from subset routes.
- +Does not dictate whether-or-not subset routes will be advertised.
- +Does NOT include Atomic-Aggregate PA

```
Rtr-3(config-router)#aggregate-address 192.168.1.0 255.255.255.0 as-set
*> 192.168.1.0 33.33.3.3 0 100 0 (65501) {222,7,99,2033,500,23} ?
*> 192.168.1.4/30 1.2.1.254 0 100 0 (65501) 222 7 99 2033 500 23 ?
*> 192.168.1.8/30 1.2.1.254 0 100 0 (65501) 222 500 23 ?
*> 192.168.1.12/30 1.2.1.254 0 100 0 (65501) 222 500 23 ?
*> 192.168.1.16/28 1.2.1.254 0 100 0 (65501) 222 7 99 2033 ?
Sw-3#
Sw-3#sho ip bgp 192.168.1.0/24
BGP routing table entry for 192.168.1.0/24, version 71
Paths: (1 available, best #1, table Default-IP-Routing-Table)
Flag: 0x820
Not advertised to any peer
(65501) {222,7,99,2033,500,23}, (aggregated by 333 33.33.3.3)
33.33.3.3 (metric 156160) from 33.33.3.3 (33.33.3.3)
Origin incomplete, metric 0, localpref 100, valid, confed-external, best
Sw-3#
```



- Must use “summary-only” keyword to suppress specific routes.
- -
- For Confederations may also select “as-confed-set” but if you select BOTH on the same line...only the LAST one you typed will take effect.





Controlling BGP Prefix Summarization With Maps



Aggregation Maps

+ Different “Maps” are available to modify the aggregated prefix.

- +Attribute-Map

- +Advertise-Map

- +Suppress-Map

+ These maps are optional, used to control characteristics of the advertised, aggregated prefix.

- +Modify Path Attributes

- +Control AS-Path info of aggregated route

- +Send aggregate route and only SOME of the subset routes.



Aggregation Maps: Attribute-Map

- + BGP Path Attributes are assigned to aggregate prefix at time of inception.
- + These PA's can either be created from scratch (i.e. Origin) or an accumulation of attributes learned from subset routes (Communities, AS-Set, etc).
- + Attribute Map used to modify/add/delete some of these attributes.



Aggregation Maps: Attribute-Map

+aggregate-address x.x.x.x <mask> attribute-map <name>

+Name of “Attribute-Map” directly correlates to name of a Route-Map where various “set” commands can change/add/remove Path Attributes of Aggregated Prefix.

+Can be used to change origin code, change/remove/add BGP Communities, add/remove Local-Preference, etc.



Aggregation Maps: Advertise-Map

+Advertise-Map

- +By default, the “aggregate-address” command encompasses all subset routes.
- +If the “as-set” or “as-confed-set” commands are used, AS-Path information from subset routes are included in the aggregate route.
- + Advertise-Map allows one to select only specific prefixes (and their associated **AS-Path** info) for consideration into the aggregated prefix.



No way to remove/suppress a specific AS from the AS-Path/AS-Set attribute.

-

This command ONLY affects AS-Path information. BGP Communities from all subset routes are still collected into the aggregate.

Example: Without Advertise-Map

```
Rtr-4#sho run | s bgp
router bgp 65502
  bgp log-neighbor-changes
  bgp confederation identifier 333
  bgp confederation peers 65501
  aggregate-address 192.168.0.0 255.255.0.0 as-set summary-only
  neighbor 33.33.1.1 remote-as 65501
  neighbor 33.33.1.1 disable-connected-check
  neighbor 33.33.1.1 update-source Loopback0
  neighbor 33.33.22.22 remote-as 65502
  neighbor 33.33.22.22 update-source Loopback0
Rtr-4#
Rtr-4#sho ip bgp | i 192.168
* > 192.168.0.0/16 0.0.0.0 100 32768 {222,7,99,2033,500,23} ?
s> 192.168.1.4/30 1.2.1.254 0 100 0 (65501) 222 7 99 2033 i
s> 192.168.1.8/30 1.2.1.254 0 100 0 (65501) 222 500 23 i
s> 192.168.1.12/30 1.2.1.254 0 100 0 (65501) 222 500 23 ?
s> 192.168.1.16/28 1.2.1.254 0 100 0 (65501) 222 7 99 2033 ?
r> 192.168.1.32/28 33.33.11.11 0 100 0 (65501 65503) i
Rtr-4#
```

Summary contains AS-Set info from ALL subset routes.

Example: **With** Advertise-Map

```
router bgp 65502
  bgp log-neighbor-changes
  bgp confederation identifier 333
  bgp confederation peers 65501
  aggregate-address 192.168.0.0 255.255.0.0 as-set summary-only advertise-map Advertise
  neighbor 33.33.1.1 remote-as 65501
  neighbor 33.33.1.1 disable-connected-check
  neighbor 33.33.1.1 update-source Loopback0
  neighbor 33.33.22.22 remote-as 65502
  neighbor 33.33.22.22 update-source Loopback0
```

```
route-map Advertise permit 10
  match ip address 1
access-list 1 permit 192.168.1.4 0.0.0.3
access-list 1 permit 192.168.1.16 0.0.0.15
```

Summary contains
AS-Set info only
from **permitted**
subset routes.

```
*> 192.168.0.0/16 0.0.0.0 100 32768 (65501) 222 7 99 2033 ?
s> 192.168.1.4/30 1.2.1.254 0 100 0 (65501) 222 7 99 2033 1
s> 192.168.1.8/30 1.2.1.254 0 100 0 (65501) 222 500 23 i
s> 192.168.1.12/30 1.2.1.254 0 100 0 (65501) 222 500 23 ?
s> 192.168.1.16/28 1.2.1.254 0 100 0 (65501) 222 7 99 2033 ?
Network Next Hop Metric LocPrf Weight Path
r> 192.168.1.32/28 33.33.11.11 0 100 0 (65501 65503) i
```

Aggregation Maps: Suppress-Map

+Objective: Transmit aggregated-prefix along with SOME of the specific prefixes.

- + Subset Routes matched in “permit” statements of ACL/Prefix-Lists/Route-Maps are “***permitted to be suppressed***” by the Suppress-Map.
- + Subset Routes not matching ACLs/Prefix-Lists/Route-Maps are advertised along with the Aggregate.
- +“Summary-Only” negated by the use of a Suppress-Map.



