



Introduction to BGP Path Attributes



BGP Path Attributes

- + Each NLRI comes with descriptive characteristics called, "Path Attributes".
- + BGP Path Attributes used to determine a variety of things:
 - + Which BGP table to insert NLRI (IPv4? IPv6?)
 - + Loop Detection
 - + Bestpath Decision algorithm
- + When receiving information about same NLRI from multiple neighbors, bestpath algorithm is used to determine which advertisement is the "best".



- Keep in mind BGP only installs the single, best path in its own local routing table
- BGP only advertises the bestpath (and all related Path Attributes) to its neighbors.

Types of BGP Path Attributes

+ Well-known attributes

- + Mandatory
- + Discretionary

+ Optional attributes

- + Transitive
- + Non-Transitive

Well-known mandatory

- AS-Path
- Next-hop
- Origin

Optional transitive

- Community
- Aggregator

Well-known discretionary

- Local preference
- Atomic aggregate

Optional non-transitive

- Multi-exit-discriminator (MED)

<http://www.iana.org/assignments/bgp-parameters/bgp-parameters.xhtml#bgp-parameters-2>



- Well-known attributes – must be supported by every BGP implementation
- Mandatory attributes – must be included with every route entry. If one attribute is missing, it will result in an error message – Ex: ORIGIN, AS_PATH, NEXT_HOP
- Discretionary attributes – every BGP router must recognize, but they don't have to be present with every route entry – Ex. Local Preference
- Optional attributes – not necessarily supported by all BGP implementations. It can be either transitive or non-transitive. – AGGREGATOR, COMMUNITY, MULTI_EXIT_DISC

BGP Path Attributes

- + Each PA describes a characteristic about the path.
- + The most common BGP path attributes are:
 - + Weight
 - + Local Preference
 - + AS_PATH
 - + Origin
 - + Multi Exit Discriminator



- Now each path attribute associated to the prefix and next-hop describe something different about the path
- The most common we see would be:
 - Weight
 - Local Preference
 - AS_PATH, which we've covered to some degree
 - Origin
 - And the Multi-exit discriminator





BGP Next-Hop Attribute



Mandatory Path Attributes – Next Hop

+ Next-Hop

- + Must be reachable via IGP routing table
- + Changed when **transmitting** eBGP updates.
- + Unchanged when **transmitting** iBGP updates.
- + IP address of eBGP neighbor when **receiving** eBGP updates
- + IP address is variable when receiving **iBGP** updates.
 - ✓ NH = eBGP peer if originally received via eBGP
 - ✓ NH = iBGP peer if locally originated by iBGP peer, or next-hop-self used.







BGP Origin-Code



Mandatory Path Attributes - Origin Code

- + Origin Code Path Attribute indicates how route was injected into BGP
 - + Via "network" statement = Origin IGP
 - ✓ Displays as "i" in BGP Table.
 - + Via redistribution = Origin Incomplete
 - ✓ Displays as "?" in BGP Table.
- + IGP > EGP > Incomplete
- + Can be changed via Route-Maps







BGP AS-Path Attribute



Mandatory Path Attributes: AS-Path

- + Indication of Autonomous Systems that have carried, and/or originated, the NLRI
- + Can be represented as;
 - + An empty list (for local-AS NLRI)
 - + An ordered, sequential list (AS-SEQ)
 - + An unordered, non-sequential list (AS-SET)
 - + An ordered, sequential list of Confederations (Confed-SEQ)
 - + An unordered, non-sequential list of Confederations (Confed-SET)
 - + A combination of of Sequence and Set information
- + Used for Loop Detection and Bestpath determination



AS Sequence

- + BGP Paths that do not contain any special characters are called "AS Sequence"
- + A sequential order of ASNs through which the BGP prefix was advertised.
- + By default, when two or more paths to same prefix have been learned, path with shortest AS-Path length is best.

```
R5#show ip bgp
```

	Network	Next Hop	Metric	LocPrf	Weight	Path
*>i	0.0.0.0	2.5.2.2	0	100	0	i
*	77.77.77.0/24	5.8.5.8	0		0	8 4056 702 899 6711 65 i
*>i		22.7.22.7	0	100	0	7 102 3004 65 i
*>i	177.77.177.0/24	22.7.22.7	0	100	0	7 8002 56001 90333 61001 i

A path with no ASNs indicates local ASN origination

Local router's BGP Peer ASN

Originating ASN

INE

AS Set

- + BGP Paths wrapped by curly braces are called an "AS Set"
- + Indicates that the route is a summarized/aggregated route
- + ASNs within the AS-Set represent an unordered list of the subnet ASNs which have been suppressed
- + Length of AS-Set not relevant to BGP Best Path Selection Algorithm

```
R5#show ip bgp
```

```
*> 77.0.0.0 22.6.22.2 0 0 12345 {7,102,3004,65,8,89,703,21} 1
```



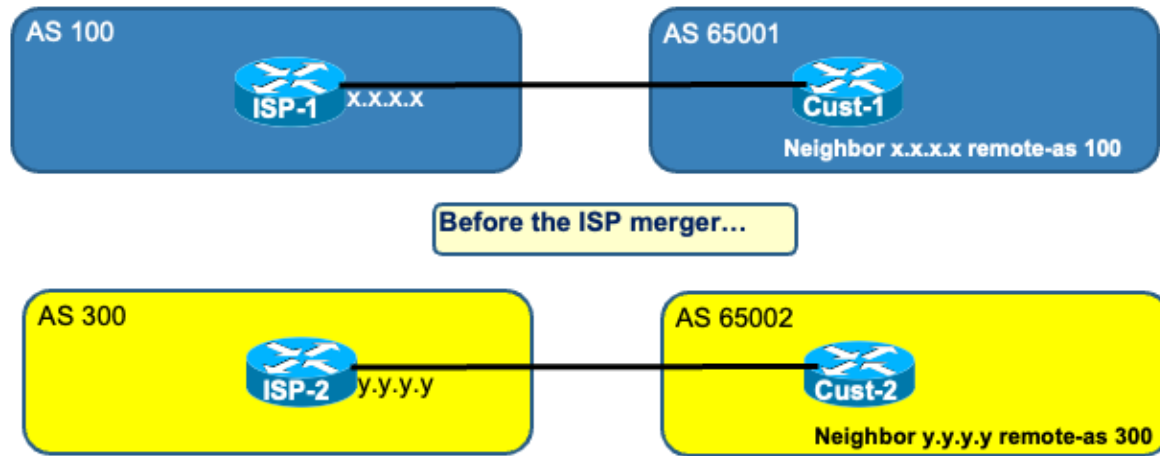
BGP Local-AS

- + The “local-as” feature allows one to change the ASN that is advertised to a peer
 - + (config-rtr)#neighbor x.x.x.x **local-as <AS#>**
- + Can only be used with eBGP peers.
- + **Prepends** local-as value to AS-Path



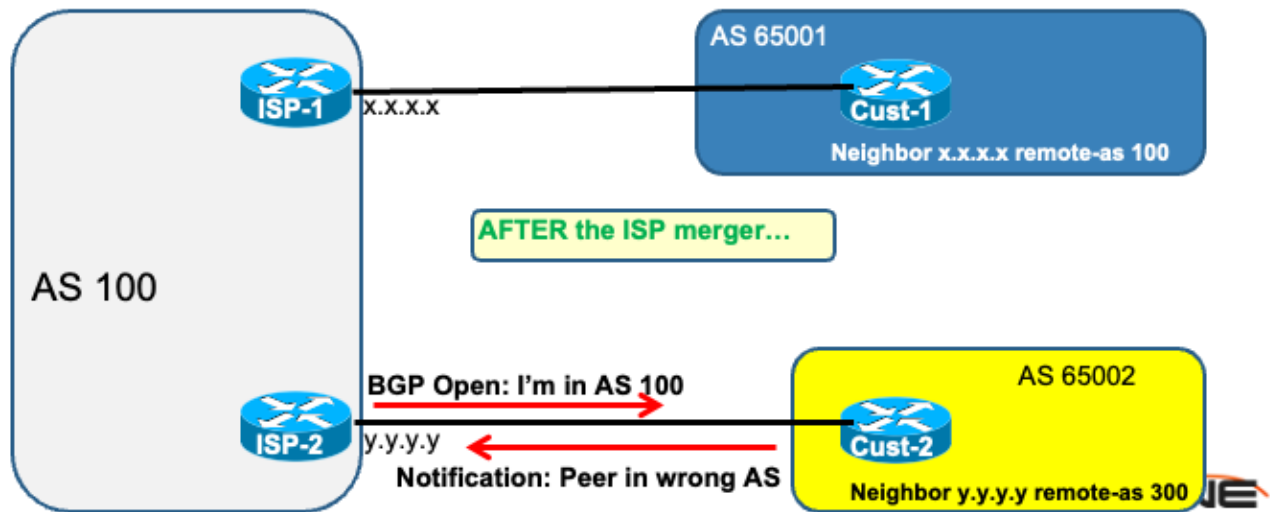
- Example from Cisco docs is that ISP-1 purchases ISP-2. If all of ISP-2's routers were reconfigured to be in ISP-1's Autonomous System this would negatively impact all of ISP-2's customers.

Local-AS Use-Case



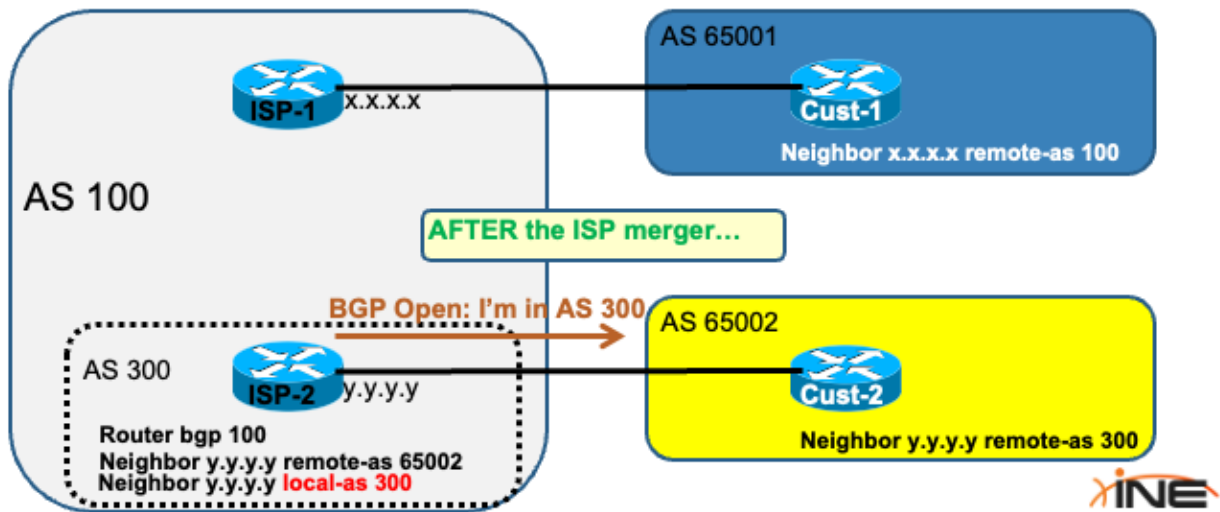
- Example from Cisco docs is that ISP-1 purchases ISP-2. If all of ISP-2's routers were reconfigured to be in ISP-1's Autonomous System this would negatively impact all of ISP-2's customers.

Why Local-AS Is Needed



- In this case, all of ISP-2's customers would need to be reconfigured. Not something you want to ask thousands of customers to do.

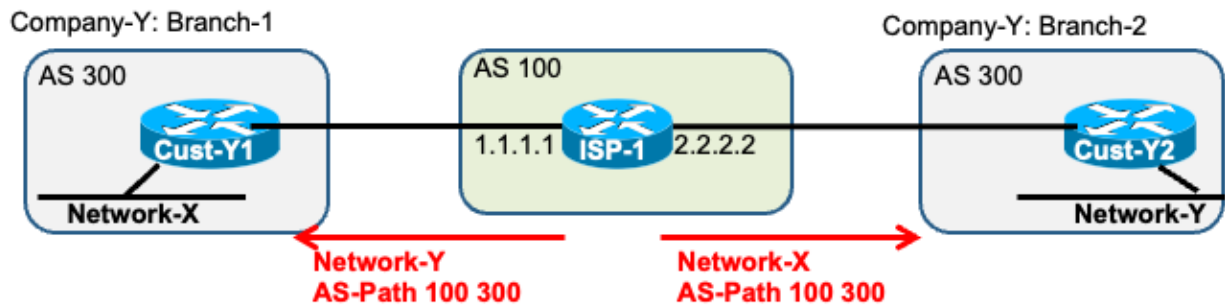
Local-AS Solves The Problem



- The initial BGP Open message sent by ISP-2 only indicates that it is in ASN-300.
- BGP Updates sent from ISP-2 to Customer-2 will have "300" prepended in front of "100".
- This was always meant to serve as a temporary measure. Eventually emails, letters etc will need to be mailed to all of the former ISPs customer informing them to change their BGP configurations.

Problems With Split ASNs

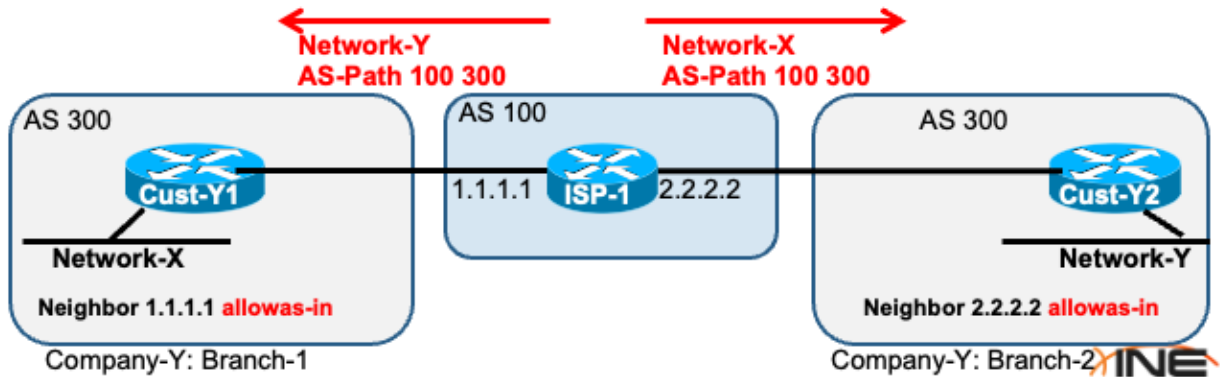
- + Due to BGP Loop Detection, neither branch will learn of the other's networks through the ISP.



- Scenario: Two branches of the same company (possibly in same city or state) connected to the same ISP. Both branches use the same Autonomous System (300).
- -
- Neither branch will learn about the other branch's networks due to AS loop detection.

BGP Allowas-In

- + BGP “allowas-in” permits installation of BGP updates containing your own, local autonomous system.
- + Only for eBGP peers.



- This is ONLY safe to implement if you know the topology VERY well and are assured there are no loops.





Cisco Route-Maps



What Problem Is Being Solved?

- + Manipulation of routing protocols often requires using some tool to:
 - + Provide route classification
 - + Manipulate/edit routing updates
- + Access-Lists and Prefix-Lists can only classify routes based on what is in the “prefix” and/or “length” fields of routing updates
- + We need a general tool that can:
 - + Be paired with ACLs or Prefix-Lists for prefix and/or mask matching
 - + Can match on other elements within routing updates
 - + Can be paired with route filtering as well as route editing features.



Introduction to Route-Maps

- + A single route-map contains one or more named, route-map entries.
- + Route-map entries are processed using If/Then/Else logic like in programming languages.
- + Each route-map entry has underlying matching parameters, configured with the **match** command.
- + To match all packets, the **route-map** clause omits the **match** command.



Route-Map Creation

- + Create initial sequence of route-map
Route-map <*descriptive name*> <permit / deny> <sequence number>
match <one-or-more items>
set <one-or-more items>
- + Repeat steps above, increasing the sequence number each time to match additional items
- + Apply Route-Map against desired feature.



Route-Map Example

```
access-list 1 permit 192.168.0.0 0.0.255.255
```

```
!
```

```
route-map Hide-Private deny 10
```

```
match ip address 1
```

```
!
```

```
Route-map Hide-Private permit 20
```

```
!
```

```
Router BGP 2000
```

```
neighbor x.x.x.x remote-as yyy
```

```
redistribute ospf 1 route-map Hide-Private
```

This route-map accomplishes the same objective as the previous example but swaps the "Permit" and "Deny" statements.

A route-map sequence with no "match" statement matches everything.



Route-Map Example

Access-list 1 **deny** 192.168.0.0 0.0.255.255

Access-list 1 permit any

!

route-map Hide-Private permit 10

match ip address 1

!

Router BGP 2000

neighbor x.x.x.x remote-as yyy

redistribute ospf 1 **route-map Hide-Private**

Anything not specifically
matched by a route-map
sequence is denied by default.

This route-map utilizes the implicit deny to
prevent sending unwanted prefixes



Route-Map Matching Logic

+ Example of IF/THEN Structure:

```
route-map TEST permit 10
```

```
  match ip address 101
```

```
  match route-type internal
```

```
route-map TEST permit 20
```

```
  match ip address 105
```

```
  match ip address 110
```

```
route-map TEST permit 30
```

When multiple "Match" statements exist beneath a single RM Sequence:

- If both statements match different criteria = logical AND
- If both statements match the same TYPE of criteria = logical OR



Route map will NOT allow you to match on BOTH a Prefix-List AND an Access-List within the same Route-Map sequence (error message).

-

Sequence 20 will actually display in running-config as "match ip address 105,110"

Translating Between Tools

(config)ip prefix-list **Payroll** Seq 10 **permit** 10.10.30.0/24

~~(config)ip prefix-list **Payroll** Seq 20 **permit** any~~

(config)route-map

(config-route-map)Match <criteria-1>

!

(config)route-map

Router <protocol>

~~Distribute-list prefix **Payroll** <in | out>~~

Distribute-list route-map **Payroll** <in | out>







Using Route-Maps with BGP



Route-Maps & BGP

- + Route-Maps can be applied in a variety of ways against a BGP configuration
- + Route-Maps can be used for classification within BGP filtering
 - + Applied against "distribute-list" statements
 - + Applied against BGP "neighbor" statements
 - + Applied against BGP "redistribute" statements
- + Route-Maps can be used to add, modify, or delete BGP path attributes
 - + Applied against BGP "neighbor" statements
 - + Applied against BGP "network" statements
 - + Applied against BGP "redistribute" statements



BGP & Route-Maps Example

```
access-list 77 permit 77.77.77.0 0.0.0.255
access-list 78 permit 77.78.78.0 0.0.0.255
```

```
route-map Attribs permit 10
  match ip address 77
  set as-path prepend 6688 123 4458 1001 3002 78009 65
!
route-map Attribs permit 20
  match ip address 78
  set as-path prepend 111 55601 333 65
!
route-map Attribs permit 30
!
```

```
router bgp 7
  bgp log-neighbor-changes
  neighbor 22.7.22.2 remote-as 12345
```



How Do I See My Changes?

- + Creating or modifying BGP policies does NOT trigger BGP to resend its routes to peers
- + There are two way to run prefixes through a new policy:
 - + Kill and restart the existing policy
 - + Router#`clear ip bgp <neighbor-ip | *>`
 - + Resets the TCP connection
 - + Utilize the BGP Route Refresh capability
 - + Router#`clear ip bgp * in`
 - + Router#`clear ip bgp * out`







BGP Prefix Classification with Prefix-Lists



The Need for Classification

- + When performing any kind of BGP NLRI filtering or manipulation, one needs to decide WHAT will be matched.
- + Several options are available:
 - + Prefix-Based Matching
 - + AS-Path Based Matching
 - + Path Attribute Matching
- + This section is about Prefix-Based Matching



Introduction To Prefix-Lists

- + Many vendors have “prefix-list” available as a feature
- + They don’t always operate in the same way
- + At a minimum, prefix-lists allow you to match on the prefix of a route
- + Some vendors (i.e. Cisco) also allow a prefix-list to match on the route length
 - + Matching on routes that have a specific subnet mask
 - + Matching on routes that have a range of subnet masks



Introduction To Cisco IP Prefix-Lists

- + IP Prefix-Lists can examine...
 - + Prefix and prefix-length
 - + Range of prefixes or range of prefix lengths.
- + IP Prefix-lists use the concept of a unique name for a single prefix-list with multiple entries.
- + IP Prefix-lists use sequence numbers to allow later addition or deletion of individual commands (entries) from a prefix-list.



Permit & Deny Usage

- + IP Prefix-lists are used to match routes, not for packet filtering
 - + **permit** keyword just implies that the route is matched
 - + **deny** keyword means the route is not matched.
- + IP Prefix-lists have a default **deny** at the end.
 - + If a **route-filtering** feature is referencing a prefix-list, any prefixes that do NOT match “permit” statements will be filtered/dropped.
 - + If a **redistribution** feature is referencing a prefix-list, any prefixes that do NOT match “permit” statements will not be redistributed.



Matching Specific Subnets

+ Cisco prefix-list matching only 10.100.46.0/24

+ (config)#ip prefix-list **INE** **permit** 10.100.46.0/24

Descriptive name

Feature referencing this prefix-list will be "permitted" to do whatever it wishes to do.

First 24-bits of prefix must exactly match 10.100.46 AND subnet-mask must exactly match /24.



Matching Ranges Of Prefixes

- + *ip prefix-list TEST seq 10 permit 10.10.0.0/16 ge 20 le 30*
 - + In this example, 10.10.0.0/16 defines the value, and quantity, of bits to be matched in the prefix.
 - + ge 20 le 30 defines the range of the prefix-length (i.e. subnet mask).
- + Match 10.10.X.X with a subnet mask between /20 and /30.



Matching Only Subnet Masks

- + *ip prefix-list INE seq 10 permit 0.0.0.0/0 ge 20 le 30*
 - + Matches any route with a subnet mask from /20 up to (and including) a /30
- + *ip prefix-list INE seq 20 permit 0.0.0.0/0 ge 32 le 32*
 - + Matches all IPv4 host routes



Operator Rules For IP Prefix-Lists

- + When both the ge and le parameters are configured, the value of the **le** parameter should be greater than or equal to the value of the **ge** parameter.
- + For example:
 - + ip prefix-list TEST seq 10 permit 10.0.0.0/8 ge 9 le 9
 - + ip prefix-list TEST seq 20 permit 172.16.0.0/16 ge 17 le 32



IP Prefix-List Example

- + `ip prefix-list TEST permit 10.0.0.0/8 ge 9`
- + `ip prefix-list TEST permit 10.0.0.0/16 le 24`
- + How is the IPv4 default route matched?
- + How are all IPv4 prefixes matched using single entry?



Path Manipulation with Route-Maps & Prefix Lists

- » Route-Maps that reference Prefix-Lists can be used for setting BGP Path Attributes.

```
ip prefix-list Test permit 150.0.0.0/8 ge 20 le 24
!
Route-map Filter permit 10
match ip address prefix Test
set local-preference 200
!
Route-map Filter permit 20
!
router bgp 10
neighbor x.x.x.x route-map Test in
```



