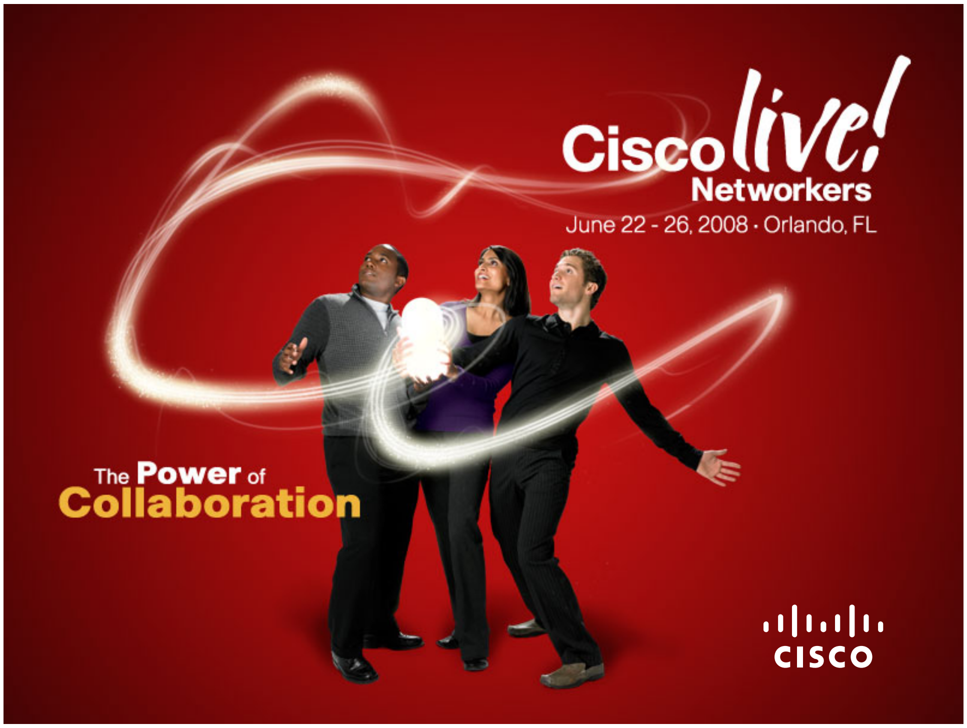
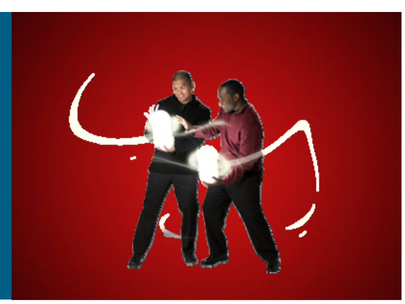


Cisco live! Networkers

Troubleshooting BGP



BRKRST-3320

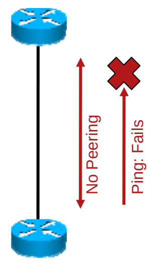
Overview

- Troubleshooting Peers
- BGP Convergence
- High Utilization
- BGP Routing Problems

Troubleshooting Peers

BGP Speakers Won't Peer

- This can be difficult to troubleshoot if you can only see one side of the connection
- Start with the simple things: check for common mistakes
  - Is it supposed to be configured for eBGP multihop?
  - Are the AS numbers right?
- Next, try pinging the peering address
  - If the ping fails, there's likely a connectivity problem



BGP Speakers Won't Peer

- Try some alternate ping options
- Is the local peering address the actual peering interface?
  - If not, use extended ping to source from the loopback or actual peering address
  - If this fails, there is an underlying routing problem
  - The other router may not know how to reach your peering interface

```
Router>enable
Router#ping
Protocol [ip]:
Target IP address: 192.168.40.1
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 172.16.23.2
```


BGP Speakers Won't Peer

- Try extended ping to sweep a range of possible MTUs
  - Note the MTU at which the ping starts to fail
  - Make certain the interface is configured for that MTU size
- If these all fail
  - None of the pings work no matter how you try....
  - It's likely a transport problem
  - Drop back and punt

```
Router>enable
Router#ping
Protocol [ip]:
Target IP address: 192.168.40.1
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface:
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp,
Verbose[none]:
Sweep range of sizes [n]: y
Sweep min size [36]: 100
Sweep max size [18024]: 2500
Sweep interval [1]: 100
....
```


BGP Speakers Won't Peer

- Remember that BGP runs on top of IP, and can be affected by:
  - Rate limiting
  - Traffic shaping
  - Tunneling problems
  - IP reachability problems (the underlying routing isn't working)
  - TCP problems
  - Etc.

BGP Speakers Won't Peer

Useful Peer Troubleshooting Commands

show tcp brief all	TCB Local Address Foreign Address (state) 64316F14 1.1.1.1.12345 2.2.2.2.179 ESTAB 6431BA8C *.179 2.2.2.2.* LISTEN 62FFDEF4 *.* *.* LISTEN
show tcp statistics	Rcvd: 7005 Total, 10 no port 0 checksum error, 0 bad offset, 0 too short .... 0 out-of-order packets (0 bytes) .... 4186 ack packets (73521 bytes) .... Sent: 9150 Total, 0 urgent packets 4810 control packets (including 127 retransmitted) 2172 data packets (71504 bytes) ....

BGP Speakers Won't Peer

Useful Peer Troubleshooting Commands

debug ip tcp transactions This can be very chatty, so be careful with this debug!	R1#sh log | i TCP0: TCP0: state was ESTAB -> FINWAIT1 [12345 -> 2.2.2.2(179)] TCP0: sending FIN TCP0: state was FINWAIT1 -> FINWAIT2 [12345 -> 2.2.2.2(179)] TCP0: FIN processed TCP0: state was FINWAIT2 -> TIMEWAIT [12345 -> 2.2.2.2(179)] TCP0: Connection to 2.2.2.2:179, advertising MSS 1460 TCP0: state was CLOSED -> SYNSENT [12346 -> 2.2.2.2(179)] TCP0: state was SYNSENT -> ESTAB [12346 -> 2.2.2.2(179)] TCP0: tcb 6430DCDC connection to 2.2.2.2:179, received MSS 1460, MSS is 1460

BGP Speakers Won't Peer

- If the connectivity is good, the next step is to check BGP itself
- `debug ip bgp`
  - Use with caution
  - Configure so the output goes to the log, rather than the console
    - `logging buffered <size>`
    - `no logging console`
  - It's easier to find the problem points this way
    - `router#show log | i NOTIFICATION`

BGP Speakers Won't Peer

- `show ip bgp neighbor 1.1.1.1 | include last reset`
  - This should give you the resets for a peer
  - The same information as is shown through `debug ip bgp`
- `bgp log-neighbor changes`
  - Provides much of the same information as `debug ip bgp`, as well

BGP Speakers Won’t Peer

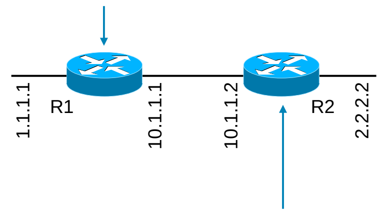
Source/Destination Address Matching

- Both sides must agree on source and destination addresses
- R1 and R2 do not agree on what addresses to use

  BGP will tear down the TCP session due to the conflict

  Points out configuration problems and adds some security

```
neighbor 2.2.2.2 remote-as 100
neighbor 2.2.2.2 update-source
loopback 0
```




```
neighbor 10.1.1.1 remote-as 100
neighbor 10.1.1.1 update-source
loopback 0
```


BGP Speakers Won’t Peer

Source/Destination Address Matching

- R2 attempts to open a session to R1

```
BGP: 10.1.1.1 open active, local address 2.2.2.2
```

- R1 denies the session because of the address mismatch
- debug ip bgp on R1 shows

```
BGP: 2.2.2.2 passive open to 10.1.1.1

BGP: 2.2.2.2 passive open failed - 10.1.1.1 is not
update-source Loopback0's address (1.1.1.1)
```

BGP Speakers Won't Peer

Active vs. Passive Peer

- Active Session

  If the TCP session initiated by R1 is the one used between R1 & R2 then R1 "actively" established the session.

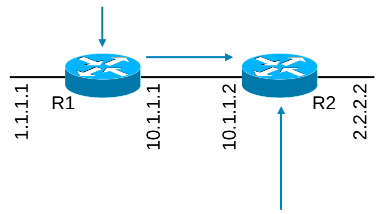
- Passive Session

  For the same scenario R2 "passively" established the session.

- R1 Actively opened the session
- R2 Passively accepted the session
- Can be configured

  ```
  neighbor x.x.x.x transport
  connection-mode
  [active|passive]
  ```

```
neighbor 2.2.2.2 remote-as 100
neighbor 2.2.2.2 connection-mode
active
```




```
neighbor 10.1.1.1 remote-as 100
neighbor 10.1.1.1 connection-mode
passive
```


BGP Speakers Won't Peer

Active vs. Passive Peer

- Use show ip bgp neighbor to determine if a router actively or passively established a session

```
R1#show ip bgp neighbors 2.2.2.2
BGP neighbor is 2.2.2.2,  remote AS 200, external link
  BGP version 4, remote router ID 2.2.2.2
[snip]
Local host: 1.1.1.1, Local port: 12343
Foreign host: 2.2.2.2, Foreign port: 179
```

- TCP open from R1 to R2's port 179 established the session
- Tells us that R1 actively established the session

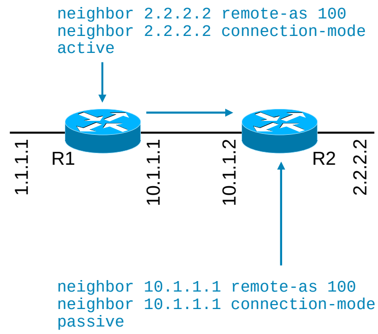
BGP Speakers Won't Peer

Session Collisions

- Both speakers initiate their sessions at the same time
- The active session established by the peer with the highest router-ID is the winner

  This rarely happens

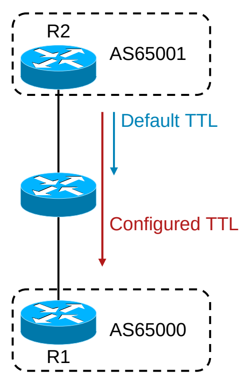
  Not an issue if this does occur



BGP Speakers Won't Peer

Time to Live

- BGP uses a TTL of 1 for eBGP peers
- For eBGP peers that are more than 1 hop away a larger TTL must be used
- neighbor x.x.x.x ebgp-multihop [2-255]




```
R1#show ip bgp neighbors 2.2.2.2 | inc External BGP
 [snip]
 External BGP neighbor may be up to 1 hops away.
```

BGP Speakers Won’t Peer

Bad Messages

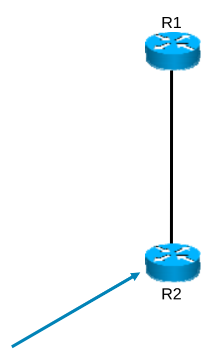
```
%BGP-3-NOTIFICATION: sent to neighbor 2.2.2.2 2/2 (peer in wrong
AS) 2 bytes 00C8 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF 002D 0104
00C8 00B4 0202 0202 1002 0601 0400 0100 0102 0280 0002 0202 00
```

unknown subcode	The peer open notification subcode isn’t known
incompatible BGP version	The version of BGP the peer is running isn’t compatible with the local version of BGP
peer in wrong AS	The AS this peer is locally configured for doesn’t match the AS the peer is advertising
BGP identifier wrong	The BGP router ID is the same as the local BGP router ID
unsupported optional parameter	There is an option in the packet which the local BGP speaker doesn’t recognize
authentication failure	The MD5 hash on the received packet does not match the correct MD5 hash
unacceptable hold time	The remove BGP peer has requested a BGP hold time which is not allowed (too low)
unsupported/disjoint capability	The peer has asked for support for a feature which the local router does not support

BGP Speaker Flap

Case Study

- Here we see a message from bgp log-neighbor-changes telling us the hold timer expired
- We can double check this by looking at `show ip bgp neighbor x.x.x.x | include last reset`




```
%BGP-5-ADJCHANGE: neighbor 10.1.1.1 Down BGP Notification sent
%BGP-3-NOTIFICATION: sent to neighbor 1.1.1.1 4/0 (hold time expired) 0
bytes
R2#show ip bgp neighbor 10.1.1.1 | include last reset
     Last reset 00:01:02, due to BGP Notification sent,hold time expired
```

BGP Speaker Flap

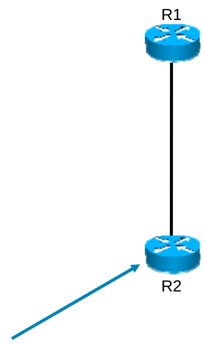
Case Study

- There are lots of possibilities here

    R1 has a problem sending keepalives?

    The keepalives are lost in the cloud?

    R2 has a problem receiving the keepalive?




```
%BGP-5-ADJCHANGE: neighbor 10.1.1.1 Down BGP Notification sent
%BGP-3-NOTIFICATION: sent to neighbor 1.1.1.1 4/0 (hold time expired) 0
bytes
R2#show ip bgp neighbor 10.1.1.1 | include last reset
     Last reset 00:01:02, due to BGP Notification sent,hold time expired
```


BGP Speaker Flap

Case Study

- Did R1 build and transmit a keepalive for R2?

```
debug ip bgp keepalive
show ip bgp neighbor
```

- When did we last send or receive data with the peer?

```
R2#show ip bgp neighbors 1.1.1.1
BGP neighbor is 1.1.1.1, remote AS 100, external link
BGP version 4, remote router ID 1.1.1.1
BGP state = Established, up for 00:12:49
Last read 00:00:45, last write 00:00:44, hold time is 180, keepalive
interval is 60 seconds
```

- If R1 did not build and transmit a KA

    How is R1 on memory?

    What is the R1's CPU load?

    Is R2's TCP window open?

BGP Speaker Flap

Case Study

```
R2#show ip bgp sum | begin Neighbor
Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  State/PfxRcd
2.2.2.2         4     2      53     284    10167    0   97 00:02:15            0
```

But the number of packets transmitted is not increasing

The number of packets generated is increasing

At least one BGP keepalive interval apart

```
R2#show ip bgp summary | begin Neighbor
Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  State/PfxRcd
2.2.2.2         4     2      53     284    10167    0   98 00:03:04            0
```

The keepalives aren't leaving R2!

BGP Speaker Flap

Case Study

- Go back to square one and check the IP connectivity

  This is a layer 2 or 3 transport issue, etc.

```
R1#ping 10.2.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/21/24 m

R1#ping ip
Target IP address: 10.2.2.2
Repeat count [5]:
Datagram size [100]: 1500
Timeout in seconds [2]:
Extended commands [n]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 1500-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

BGP Convergence

BGP Slow Convergence

- Hey—Who are you calling slow?
  - Slow is a relative term....
  - BGP probably won't ever converge as fast as any of the IGPs
- Two general convergence situations
  - Initial startup between peers
  - Route changes between existing peers

BGP Slow Convergence

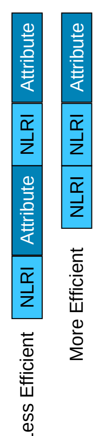
Initial Convergence

- Initial convergence is limited by
  - The number of packets required to transfer the entire BGP database
    - The number of routes
    - The ability of BGP to pack routes into a small number of packets
    - The number of peer specific policies
  - TCP transport issues
    - How often does TCP go into slow start?
    - How much can TCP put into one packet?

BGP Slow Convergence

Initial Convergence

- BGP starts a packet by building an attribute set
- It then packs as many destinations (NLRIs) as it can into the packet
  - Only destinations with the same attribute set can be placed in the packet
  - Destinations can only be put into the packet until it's full
- First rule of thumb: to increase convergence speed, decrease unique sets of attributes



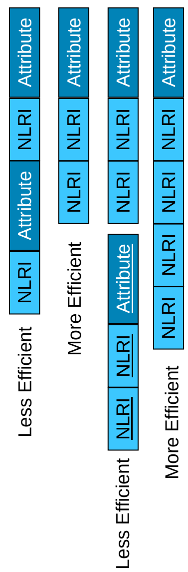
BGP Slow Convergence

Initial Convergence

- The larger the packet BGP can build, the more destinations it can put in the packet

  The more you can put in a single packet, the less often you have to repeat the same attributes

  Second rule of thumb: allow BGP to use the largest packets possible

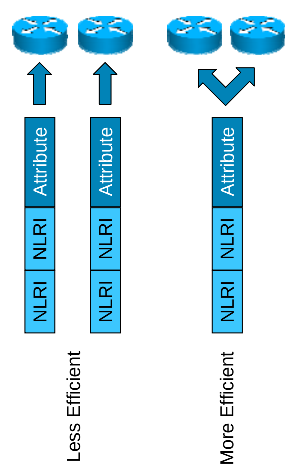


BGP Slow Convergence

Initial Convergence

- BGP must create packets based the policies towards each peer

  Third rule of thumb: Minimize the number of unique policies towards eBGP peers



BGP Slow Convergence

Initial Convergence

- TCP Interactions
  - Each time a TCP packet is dropped, the session goes into slow start
  - It takes a good deal of time for a TCP session to come out of slow start
  - Fourth rule of Thumb: Try and reduce the circumstances under which a TCP segment will be dropped during initial convergence

BGP Slow Convergence

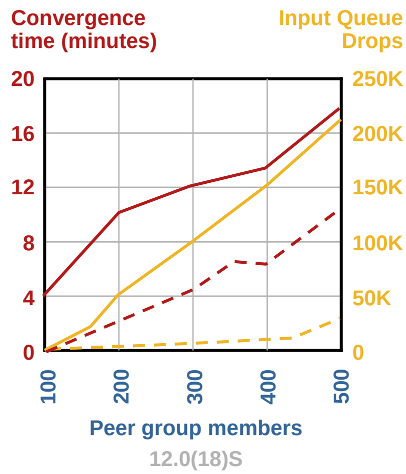
Initial Convergence

- Bottom Line:
  - Hold down the number of unique attributes per route
    - Don't send communities if you don't need to, etc
  - Hold down the number of policies towards eBGP peers
    - Try to find a small set of common policies, rather than individualizing policies per peer
  - Stop TCP segment drops
    - Increase input queues
    - Increase SPD thresholds
    - Make certain links are clean

BGP Slow Convergence

Initial Convergence

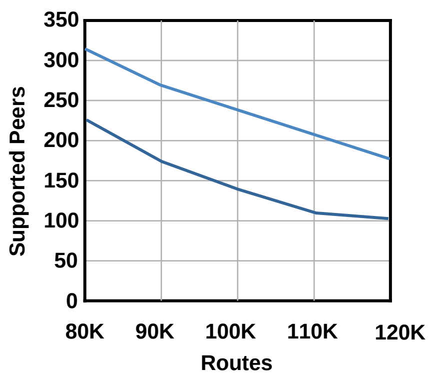
- Here we see the results of setting up maximum sized input queues
    - A single router running 12.0(18)S
    - 100 to 500 peers in a single peer group
    - Sending 100,000+ routes to each peer
- Increasing the input queue sizes
    - Reduced the input queue drops by a factor of 10
    - Reduces convergence speed by 50%



BGP Slow Convergence

Initial Convergence

- TCP MTU path discovery allows BGP to use the largest packets possible
- Without PMTU discovery, we can support 100 peers with 120,000 routes each
- With PMTU discover, we can support 175 peers with 120,000 routes each
- Note this is 12.0(18)S, Cisco IOS Software can support more than this now!



BGP Slow Convergence

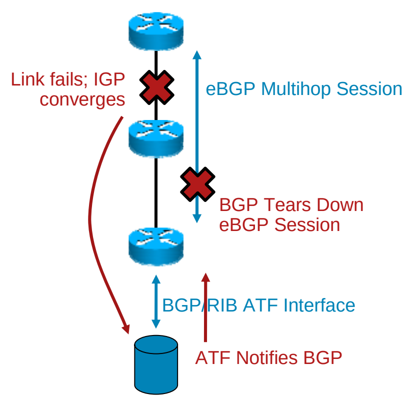
Route Change Convergence

- There are two elements to route change convergence for BGP
  - How long does it take to see the failure?
  - How long does it take to propagate information about the failure?
- For faster peer down detection, there are several tools you can use
  - Fast layer two down detection
  - Fast external fallover for directly connected eBGP peers
  - Faster keepalive and dead interval timers
    - Down to 3 and 9 are commonly used today

BGP Slow Convergence

Route Change Convergence

- Fast Session Deactivation
  - The address of each peer is registered with the Address Tracking Filter (ATF) system
  - When the state of the route changes, ATF notifies BGP
  - BGP tears down the peer impacted
  - BGP does not wait on the hold timer to expire



BGP Slow Convergence

Route Change Convergence

- Very dangerous for iBGP peers
  - IGP may not have a route to a peer for a split second
  - FSD would tear down the BGP session
  - Imagine if you lose your IGP route to your RR (Route Reflector) for just 100ms
- Off by default
  - `neighbor x.x.x.x fall-over`

BGP Slow Convergence

Route Change Convergence

- ATF can also be used to track changes in next hops
  - iBGP recurses onto an IGP next hop to find a path through the local AS
  - Changes in the IGP cost or reachability are normally seen only by the BGP scanner
  - Since the scanner runs every 60 seconds, by default, this means iBGP convergence can take up to 60 seconds on an IGP change....

BGP Slow Convergence

Route Change Convergence

- BGP Next Hop Tracking
  - Enabled by default
  - `[no] bgp nexthop trigger enable`
- BGP registers all nexthops with ATF
  - Hidden command will let you see a list of nexthops
  - `show ip bgp attr nexthop`
- ATF will let BGP know when a route change occurs for a nexthop
- ATF notification will trigger a lightweight “BGP Scanner” run
  - Bestpaths will be calculated
  - None of the other “Full Scan” work will happen

BGP Slow Convergence

Route Change Convergence

- Once an ATF notification is received BGP waits 5 seconds before triggering NHT scan
  - `bgp nexthop trigger delay <0-100>`
  - May lower default value as we gain experience
- Event driven model allows BGP to react quickly to IGP changes
  - No longer need to wait as long as 60 seconds for BGP to scan the table and recalculate bestpaths
  - Tuning your IGP for fast convergence is recommended

BGP Slow Convergence

Route Change Convergence

- Dampening is used to reduce frequency of triggered scans
- show ip bgp internal
  - Displays data on when the last NHT scan occurred
  - Time until the next NHT may occur (dampening information)
- New commands
  - `bgp nexthop trigger enable`
  - `bgp nexthop trigger delay <0-100>`
  - `show ip bgp attr next-hop ribfilter`
  - `debug ip bgp events nexthop`
  - `debug ip bgp rib-filter`
- Full BGP scan still happens every 60 seconds
  - Full scanner will no longer recalculate bestpaths if NHT is enabled

BGP Slow Convergence

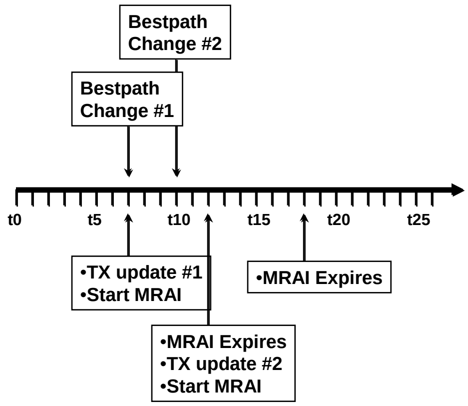
Route Change Convergence

- How is the timer enforced for peer X?
  - Timer starts when all routes have been advertised to X
  - For the next MRAI (seconds) we will not propagate any bestpath changes to peer X
  - Once X's MRAI timer expires, send him updates and withdraws
  - Restart the timer and the process repeats…
- User may see a wave of updates and withdraws to peer X every MRAI
- User will NOT see a delay of MRAI between each individual update and/or withdraw
  - BGP would probably never converge if this was the case

BGP Slow Convergence

Route Change Convergence

- MRAI timeline for iBGP peer
- Bestpath Change #1 at t7 is TXed immediately
- MRAI timer starts at t7, will expire at t12
- Bestpath Change #2 at t10 must wait until t12 for MRAI to expire
- Bestpath Change #2 is TXed at t12
- MRAI timer starts at t12, will expire at t17
- MRAI expires at t17…no updates are pending



BGP Slow Convergence

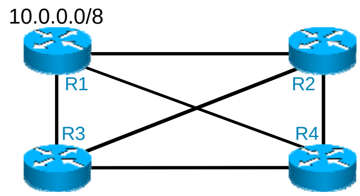
Route Change Convergence

- BGP is not a link state protocol
- May take several “rounds/cycles” of exchanging updates and withdraws for the network to converge
- MRAI must expire between each round!
- The more fully meshed the network and the more tiers of ASes, the more rounds required for convergence
- Think about
  - How many tiers of ASes there are in the Internet
  - How meshy peering can be in the Internet

BGP Slow Convergence

Route Change Convergence

- Full mesh is the worst case MRAI convergence scenario
- R1 will send a withdraw to all peers for 10.0.0.0/8
- Count the number of rounds of UPDATEs and withdraws until the network has converged
- Note how MRAI slows convergence
- Blue path is the bestpath

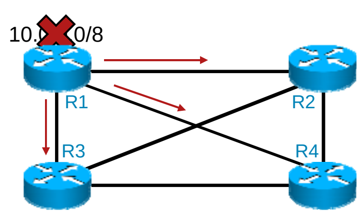


R2	R1	R3,R1	R4,R1	
R3	R1	R2,R1	R4,R1	
R4	R1	R2,R1	R3,R1	

BGP Slow Convergence

Route Change Convergence

- R1 withdraws 10.0.0.0/8 to all peers
- R1 starts a MRAI timer for each peer



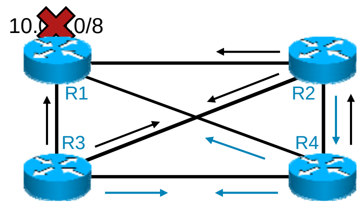
R2	~~R1~~	R3,R1	R4,R1	
R3	~~R1~~	R2,R1	R4,R1	
R4	~~R1~~	R2,R1	R3,R1	

Withdraw
Denied Update
Update

BGP Slow Convergence

Route Change Convergence

- R2, R3, & R4 recalculate their bestpaths
- R2, R3, & R4 send updates based on new bestpaths
- R2, R3, & R4 start a MRAI timer for each peer
- End of Round 1



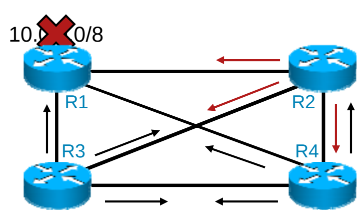
R2	~~R1~~	R3,R1	R4,R1	
R3	~~R1~~	R2,R1	R4,R1	
R4	~~R1~~	R2,R1	R3,R1	

Withdraw
Denied Update
Update

BGP Slow Convergence

Route Change Convergence

- R2, R3, & R4 recalculate their bestpaths
- R2, R3 & R4 must wait for their MRAI timers to expire!
- R2, R3, & R4 send updates and withdraws based on their new bestpaths
- R2, R3, & R4 restart the MRAI timer for each peer
- End of Round 2



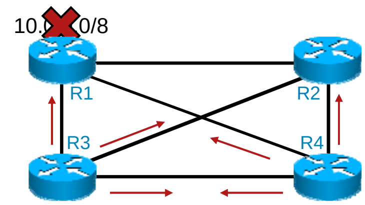
R2	~~R1~~	~~R3,R1~~	~~R4,R1~~	
R3	~~R1~~	~~R2,R1~~	R4,R2,R1	
R4	~~R1~~	~~R2,R1~~	R3,R2,R1	R2,R3,R1

Withdraw
Denied Update
Update

BGP Slow Convergence

Route Change Convergence

- R3 & R4 recalculate their bestpaths
- R3 & R4 must wait for their MRAI timers to expire!
- R3 & R4 send updates and withdraws based on their new bestpaths
- R3 & R4 restart the MRAI timer for each peer
- End of Round 3



R2	~~R1~~	~~R2,R1~~	~~R4,R1~~	
R3	~~R1~~	~~R2,R1~~	~~R4,R2,R1~~	
R4	~~R1~~	~~R2,R1~~	~~R3,R2,R1~~	~~R2,R3,R1~~

Withdraw
Denied Update
Update

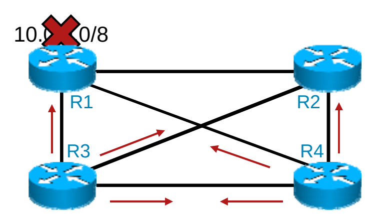
BGP Slow Convergence

Route Change Convergence

- R2, R3, & R4 took 3 rounds of messages to converge
- MRAI timers had to expire between 1st/2nd round and between 2nd/3rd round
- Total MRAI convergence delay for this example

  iBGP mesh – 10 seconds

  eBGP mesh – 60 seconds



R2	~~R1~~	~~R2,R1~~	~~R4,R1~~	
R3	~~R1~~	~~R2,R1~~	~~R4,R2,R1~~	
R4	~~R1~~	~~R2,R1~~	~~R3,R2,R1~~	~~R2,R3,R1~~

Withdraw
Denied Update
Update

BGP Slow Convergence

Route Change Convergence

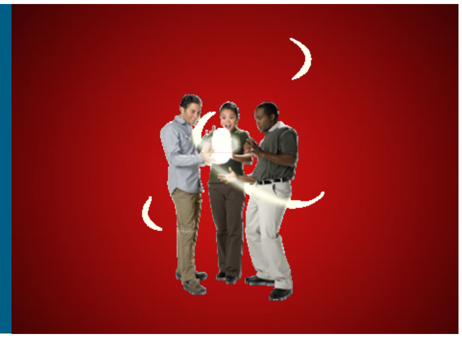
- Internet churn means we are constantly setting and waiting on MRAI timers
  - One flapping prefix slows convergence for all prefixes
  - Internet table sees roughly 6 bestpath changes per second
- For iBGP and PE-CE eBGP peers
  - `neighbor x.x.x.x advertisement-interval 0`
  - Will be the default in 12.0(32)S
- For regular eBGP peers
  - Lowering to 0 may get you dampened
  - OK to lower for eBGP peers if they are not using dampening

BGP Slow Convergence

Route Change Convergence

- Will a MRAI of 0 eliminate batching?
  - Somewhat but not much happens anyway
  - TCP, the operating system, and BGP code provide some batching
    - Process all message from peer InQs
    - Calculate bestpaths based on received messages
    - Format UPDATEs to advertise new bestpaths
- What about CPU load from 0 second MRAI?
  - Internet table has ~6 bestpath changes per second
  - Easy for a router to handle, 5 seconds of delay is not needed

High Utilization



High Utilization

- High Processor Utilization
- Next Hop Tracking
- High Memory Utilization

High Processor Utilization

- Why?
  - This could be for several reasons
  - High route churn is the most likely

```
router# show process cpu
  CPU utilization for five seconds: 100%/0%; one minute: 99%; five minutes:
81%
....
139      6795740   1020252       6660 88.34% 91.63% 74.01%   0 BGP Router
```


High Processor Utilization

- Check how busy the peers are
  - The Table Version
    - You have 150k routes and see the table version increase by 150k every minute … something is wrong
    - You have 150k routes and see the table version increase by 300 every minute … sounds like normal network churn
  - The InQ
    - Flood of incoming updates or build up of unprocessed updates
  - The OutQ
    - Flood of outgoing updates or build up of untransmitted updates

```
router# show ip bgp summary
  Neighbor   V     AS MsgRcvd MsgSent  TblVer  InQ  OutQ  Up/Down State/PfxRcd
  10.1.1.1   4  64512  309453  157389  19981     0   253 22:06:44 111633
  172.16.1.1 4  65101  188934    1047  40081    41     0 00:07:51 58430
```

High Processor Utilization

- If the Table Version is Changing Quickly
  - Are you in initial convergence with this peer?
  - Is the peer flapping for some reason?
  - Examine the table entries from this peer: why are they changing?
  - If there is a group of routes which are constantly changing, consider route flap dampening
- If the InQ is high
  - You should see the table version changing quickly
  - If it's not, the peer isn't acting correctly
  - Consider shutting it down until the peer can be fixed
- If the OutQ is high
  - Lots of updates being generated
  - Check table versions of other peers
  - Check for underlying transport problems

High Processor Utilization

- Check on the BGP Scanner
  - Walks the table looking for changed next hops
  - Checks conditional advertisement
  - Imports from and exports to VPNv4 VRFs

```
router# show processes | include BGP Scanner
 172 Lsi 407A1BFC       29144     29130    1000 8384/9000  0 BGP Scanner
```

High Processor Utilization

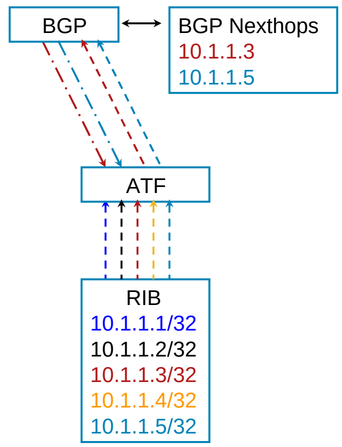
- To relieve pressure on the BGP Scanner
  - Upgrade to newer code
    - Most of the work of the BGP Scanner has been moved to an event driven model
    - This has reduced the impact of BGP Scanner significantly
  - Reduce route and view count
  - Reduce or eliminate other processes which walk the RIB
    - SNMP routing table walks, for instance
  - Deploy BGP Next Hop Tracking (NHT)

Next Hop Tracking

- ATF is a middle man between the RIB and RIB clients
  - BGP, OSPF, EIGRP, etc are all clients of the RIB
- A client tells ATF what prefixes he is interested in
- ATF tracks each prefix
  - Notify the client when the route to a registered prefix changes
  - Client is responsible for taking action based on ATF notification
  - Provides a scalable event driven model for dealing with RIB changes

Next Hop Tracking

- BGP tells ATF to let us know about any changes to 10.1.1.3 and 10.1.1.5
- ATF filters out any changes for 10.1.1.1/32, 10.1.1.2/32, and 10.1.1.4/32
- Changes to 10.1.1.3/32 and 10.1.1.5/32 are passed along to BGP



Next Hop Tracking

- BGP Next Hop Tracking
  - Enabled by default
  - `[no] bgp nexthop trigger enable`
- BGP registers all nexthops with ATF
  - Hidden command will let you see a list of nexthops
  - `show ip bgp attr nexthop`
- ATF will let BGP know when a route change occurs for a nexthop
- ATF notification will trigger a lightweight “BGP Scanner” run
  - Bestpaths will be calculated
  - None of the other “Full Scan” work will happen

Next Hop Tracking

- Once an ATF notification is received BGP waits 5 seconds before triggering NHT scan

  `bgp nexthop trigger delay <0-100>`

  May lower default value as we gain experience

- Event driven model allows BGP to react quickly to IGP changes

  No longer need to wait as long as 60 seconds for BGP to scan the table and recalculate bestpaths

  Tuning your IGP for fast convergence is recommended

Next Hop Tracking

- Dampening is used to reduce frequency of triggered scans
- show ip bgp internal

  Displays data on when the last NHT scan occurred

  Time until the next NHT may occur (dampening information)

- New commands

```
bgp nexthop trigger enable
bgp nexthop trigger delay <0-100>
show ip bgp attr next-hop ribfilter
debug ip bgp events nexthop
debug ip bgp rib-filter
```

- Full BGP scan still happens every 60 seconds

  Full scanner will no longer recalculate bestpaths if NHT is enabled

High Memory Utilization

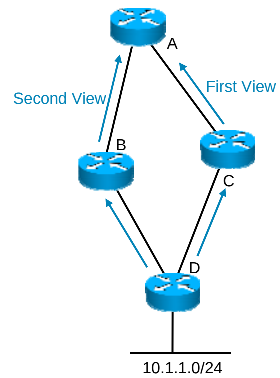
Views and Routes

- Why is BGP taking up so much memory?

  A BGP speaker generally receives a number of copies of the same route or set of routes

  Each of these copies of the same route or routes is called a view

  A has two views of 10.1.1.0/24



High Memory Utilization

Views and Routes

- Multiple views can come from:

  iBGP peers peering with the same remote AS

  iBGP peers peering with remote AS' with (generally) the same table

    This is common in the case of the global Internet

  eBGP peers peering with the same remote AS

  eBGP peers peering with remote AS' with (generally) the same table

    This is common in the case of the global Internet

High Memory Utilization

Views and Routes

- Multiple views exist in IGPs, as well
  - But not on the same scale
  - Neighbor adjacencies in IGPs are generally on a lower scale
    - In the hundreds, not the thousands
  - Neighbor adjacencies in IGPs normally pick up different routes, rather than the same route multiple times
- Each view takes up some amount of space
  - 250,000 routes x 100 views == a lot of memory usage

High Memory Utilization

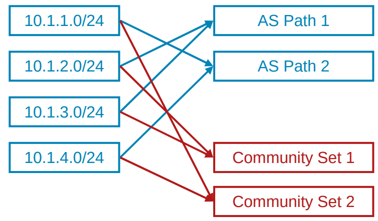
Views and Routes

- To reduce memory consumption:
  - Reduce the number of routes
    - This is particularly true in providers supporting L3VPN services
    - The route and view count can escalate quickly when supporting many customer's L3VPNs
  - Filter aggressively
  - Accept partial routing tables, rather than full routing tables
  - Reduce the number of views
    - Use route reflectors rather than full mesh iBGP peering
    - Peer only when needed

High Memory Utilization

Attributes

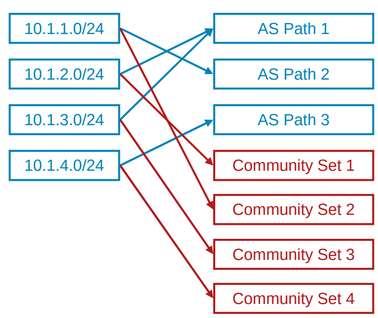
- BGP implementations build their memory structures around minimizing storage
- Attributes are stored once
  - Rather than once per route
  - Each route references an attribute set, rather than storing the attribute set
- This is similar to the way BGP updates are formed



High Memory Utilization

Attributes

- The more unique attribute sets you're receiving, the more unique attribute sets you need to store
  - You might have the same number of routes and views over time, but memory utilization can increase



High Memory Utilization

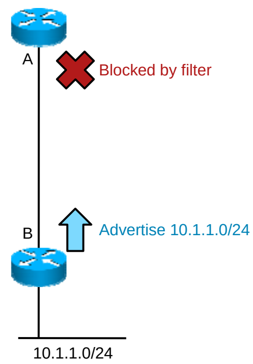
Attributes

- To Conserve Memory
  - Strip unneeded attributes on the inbound side of eBGP peering sessions
    - Verify you don't really need them, or they aren't useful after the route has transited your AS
    - Communities are the biggest/only target
  - Use Communities wisely within your network
    - A large mishmash of communities can consume memory

High Memory Utilization

Soft Reconfiguration

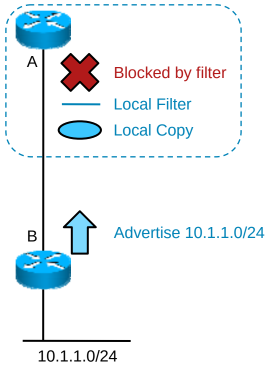
- B advertises 10.1.1.0/24 to A
- A filters the route locally
- The filters on A are changed to permit 10.1.1.0/24
- But how does A relearn 10.1.1.0/24?



High Memory Utilization

Soft Reconfiguration

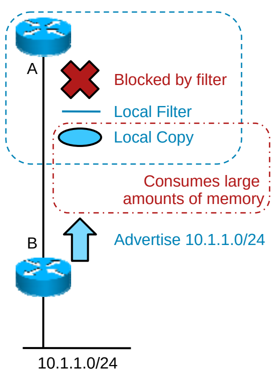
- With soft reconfiguration, A saves all the routes it receives from B
- Applies any inbound filters between this saved copy of B's updates and the local BGP table
- If the local filters change, they can be reapplied by simply pulling all the updates from the saved table into the local BGP table



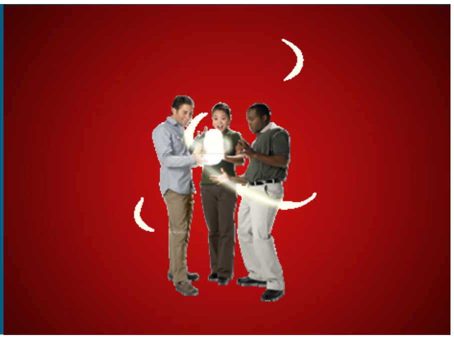
High Memory Utilization

Soft Reconfiguration

- Keeping this local copy uses a lot of memory
- In general, don't use soft-reconfiguration
- BGP now uses the route refresh capability to rebuild the local table if the local filters change



Routing Problems

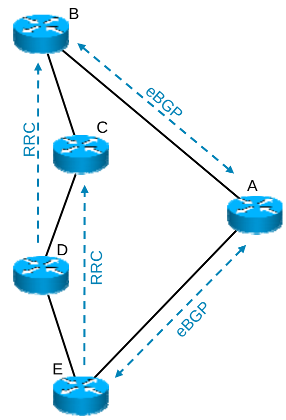


BGP Routing Problems

- Route Reflector Loops
- Route Reflector Suboptimal Routes
- Inbound Traffic Path Problems

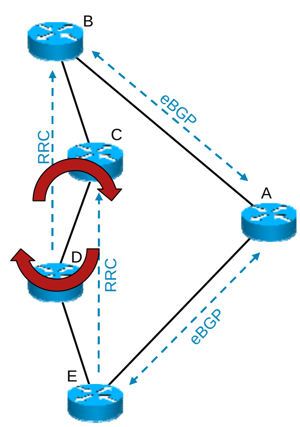
Route Reflector Loops

- Router B
  - BGP Next-Hop: Router A
  - Local Next-Hop: Router A
  - Set: Next-Hop-Self
- Router C
  - BGP Next-Hop: Router B
  - Local Next-Hop: Router D
- Router D
  - BGP Next-Hop: Router E
  - Local Next-Hop: Router C
- Router E
  - BGP Next-Hop: Router A
  - Local Next-Hop: Router A
  - Set: Next-Hop-Self



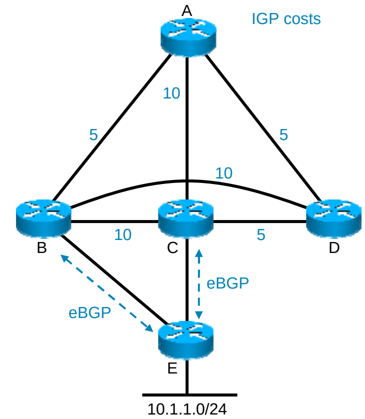
Route Reflector Loops

- This results in a permanent routing loop
- Route reflectors must always follow the topology
- Never peer through a route reflector client to reach a route reflector



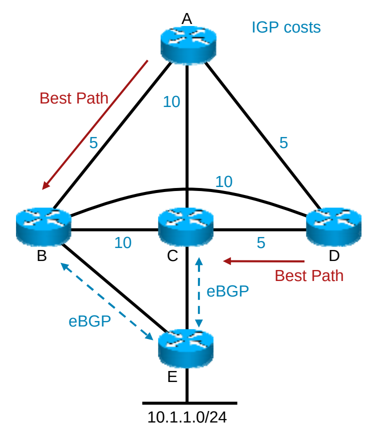
Route Reflector Suboptimal Routing

- Route reflectors can also cause routing to be different (or suboptimal) compared to full mesh iBGP
- E advertises 10.1.1.0/24 through eBGP to both B and C
- The local preference, MED, AS Path length, and all other attributes are the same for 10.1.1.0/24 at both B and C



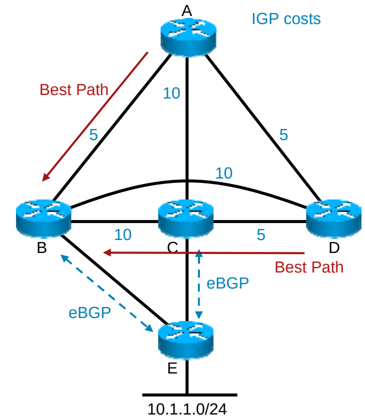
Route Reflector Suboptimal Routing

- Assume A, B, C, and D are configured for full mesh iBGP
- A chooses B as its exit point because of the IGP cost
- D chooses C as its exit point, because of the IGP cost



Route Reflector Suboptimal Routing

- Assume B, C and, D are configured as route reflector clients of A
- A chooses B as its best path because of the IGP cost
- A reflects this choice to C, but C chooses its locally learned eBGP route over the internal through B
- A reflects this choice to D, and D chooses the path through B, even though the path through C is shorter



Route Reflector Suboptimal Routing

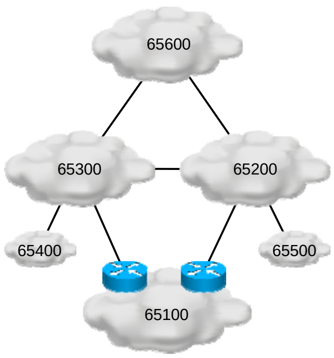
- There is little you can do about this
- Whenever you remove routing information, you risk suboptimal routing
- Keeping the route reflector topology in line with the layer 3 topology helps
- iBGP multipath can resolve some of these problems
  - At the cost of additional memory
- Otherwise, use policy to choose the best exit point

Impacting Inbound Traffic Path

- I'm in AS65100
- Why does my traffic

    Come in through AS65200 and AS65300, although I want it to come in through AS65300 only?

    Even though I do AS Path Prepend....



Impacting Inbound Traffic Path

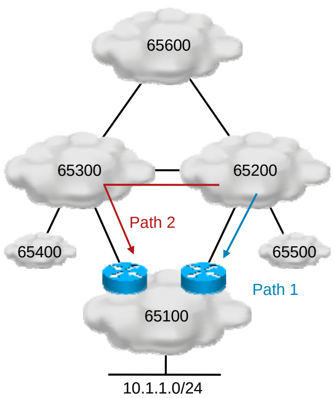
- Why would AS65200 ever prefer Path 2 over Path 1

    You pay for the AS65200 link

    They pay for the AS65200 to AS65300 link

    If they preferred Path 2, they would be paying to support your preferred inbound traffic path

    There's not much of a chance of this happening....



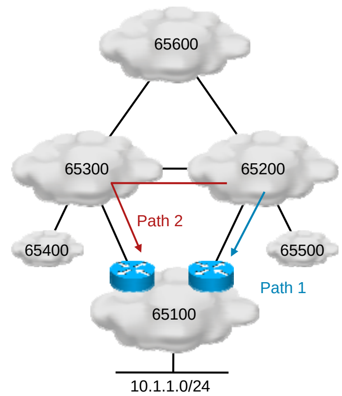
Impacting Inbound Traffic Path

- How does AS65200 implement this policy?

  Routes received from customers are preferred over routes received from peers, using Local Preference

  Adding AS Path hops won't overcome AS65200's Local Preference

  So, traffic from AS65500 will always come in through the AS65200 link, as long as you're advertising 10.1.1.0/24 through the link



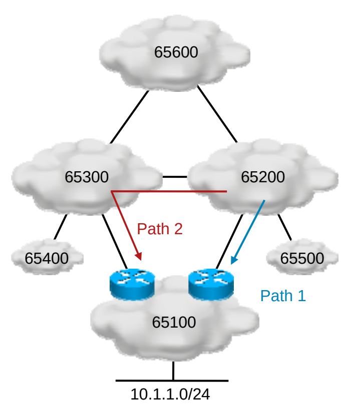
Impacting Inbound Traffic Path

- Possible Solutions

  Live with traffic from AS65200's peers coming in through this link

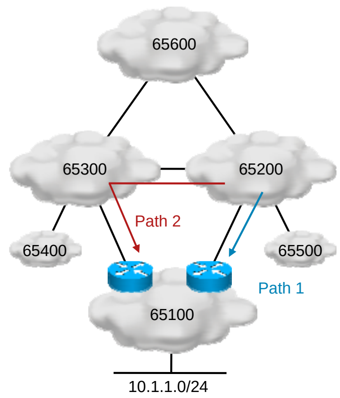
  Use conditional advertisement

    Conditional advertisement could be slow, though



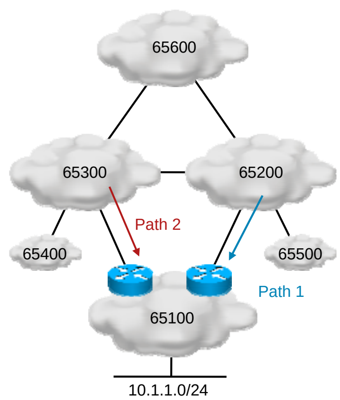
Impacting Inbound Traffic Path

- Possible Solutions
  - Use RFC1998 Communities
    - You set a community on 10.1.1.0/24
    - AS65200 translates this community into a Local Preference
    - AS65200 then prefers the route through AS65300 over the connected route
    - Don't count on this happening—most providers don't support RFC1998 communities



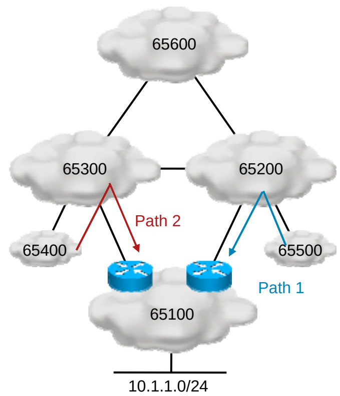
Impacting Inbound Traffic Path

- Why can't I load share traffic between the two links?
  - I've tried AS Path prepend, why doesn't it work?



Impacting Inbound Traffic Path

- Any traffic from AS65500 will always come through AS65200
- Any traffic from AS65300 will always come through AS65300
- There's no way to alter this
- So, if the majority of your traffic comes from AS65500, there's not much you can do....



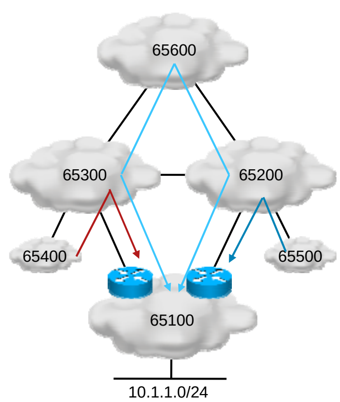
Impacting Inbound Traffic Path

- The only traffic you can really adjust with AS Path prepend is from AS65600

  You can influence which path AS65600 will take

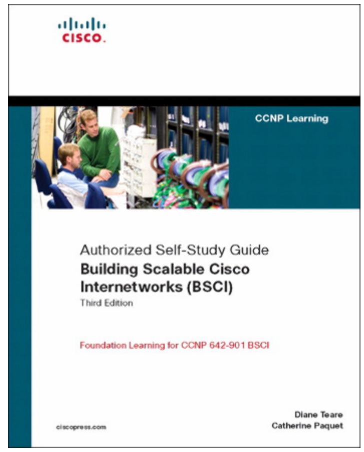
  Through AS65200 or through AS65200

  This may or may not allow you to tune inbound traffic well



Recommended Reading

- Continue your Cisco Live learning experience with further reading from Cisco Press
- Check the Recommended Reading flyer for suggested books



Available Onsite at the Cisco Company Store

Complete Your Online Session Evaluation

- Give us your feedback and you could win fabulous prizes. Winners announced daily.
- Receive 20 Passport points for each session evaluation you complete.
- Complete your session evaluation online now (open a browser through our wireless network to access our portal) or visit one of the Internet stations throughout the Convention Center.

Don't forget to activate your Cisco Live virtual account for access to all session material on-demand and return for our live virtual event in October 2008.

Go to the Collaboration Zone in World of Solutions or visit www.cisco-live.com.