BGP Scaling Techniques

UNIVERSITY OF OREGON

Last updated 5th May 2026

NSRC
Network Startup Resource Center

Acknowledgements

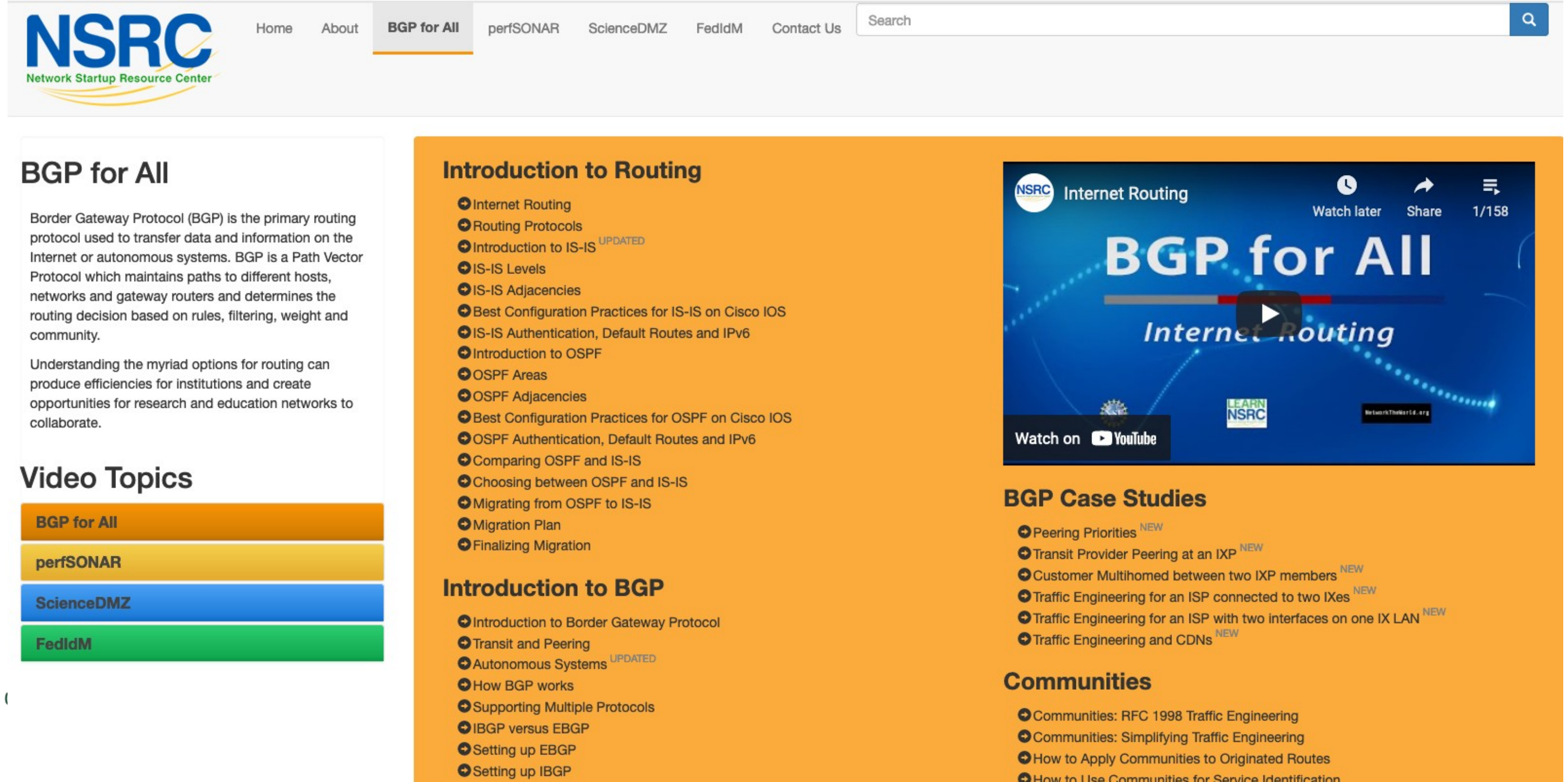
- This material originated from the Cisco ISP/IXP Workshop Programme developed by Philip Smith & Barry Greene

- Use of these materials is encouraged as long as the source is fully acknowledged, and this notice remains in place

- Bug fixes and improvements are welcomed
  - Please email learn (at) nsrc (dot) org

BGP Videos

- NSRC has made a video recording of this presentation, as part of a library of BGP videos for the whole community to use:
  - https://learn.nsrc.org/bgp#bgp_scaling_techniques



BGP Scaling Techniques

- Original BGP specification and implementation was fine for the Internet of the early 1990s
  - But didn't scale
- Issues as the Internet grew included:
  - Scaling the IBGP mesh beyond a few peers?
  - Implement new policy without causing flaps and route churning?
  - Keep the network stable, scalable, as well as simple?

BGP Scaling Techniques

- BGP Configuration Scaling
  - Grouping BGP peers
- Industry Best Practice Scaling Techniques
  - Route Refresh
  - Route Reflectors
- Historical Scaling Techniques
  - Soft Reconfiguration
  - Confederations
  - Route Flap Damping

BGP Configuration Scaling

Cisco’s peer-groups

&

Juniper’s BGP groups

Grouping similar BGP peers

- What are they for?
  - Lets operators group peers with the same outbound policy
  - Makes configuration easier
  - Makes configuration less prone to error
  - Makes configuration more readable
  - Members can have different inbound policy
  - Can be used for EBGP neighbours too!

Grouping similar BGP peers

- Cisco:
  - peer-groups
    - Originally designed to speed IBGP convergence – now for scaling BGP configuration management
  - Internal code optimisation called *update-groups*
    - Speeds IBGP convergence; update only calculated once for neighbours with the same outbound policy
- Juniper:
  - BGP groups

Configuring a Peer Group in IOS

```
router bgp 64500
 address-family ipv4
  neighbor IBGP peer-group
  neighbor IBGP remote-as 64500
  neighbor IBGP update-source loopback 0
  neighbor IBGP send-community
  neighbor IBGP route-map outfilter out
  neighbor 100.64.0.1 peer-group IBGP
  neighbor 100.64.0.2 peer-group IBGP
  neighbor 100.64.0.2 route-map  infilter in
  neighbor 100.64.0.3 peer-group IBGP
!
```

Note how 100.64.0.2 has an additional inbound filter over the peer-group

Configuring a Peer Group in IOS

```
router bgp 64500
 address-family ipv4
  neighbor EBGP peer-group
  neighbor EBGP send-community
  neighbor EBGP route-map set-metric out
  neighbor 100.89.1.2 remote-as 64502
  neighbor 100.89.1.2 peer-group EBGP
  neighbor 100.89.1.4 remote-as 64503
  neighbor 100.89.1.4 peer-group EBGP
  neighbor 100.89.1.6 remote-as 64504
  neighbor 100.89.1.6 peer-group EBGP
  neighbor 100.89.1.6 filter-list infilter in
!
```

Can be used for EBGP as well

Peer Groups

- Peer-groups are considered obsolete by Cisco:
  - Replaced by update-groups (internal coding – not configurable)
- But are still considered best practice by many network operators
- Cisco introduced peer-templates
  - A much-enhanced version of peer-groups, allowing more complex constructs

Cisco's update-groups (1)

- Update-groups is an internal IOS coding, taking over the performance gains introduce by peer-groups

```
Router1#sh ip bgp 10.0.0.0/26
BGP routing table entry for 10.0.0.0/26, version 2
Paths: (1 available, best #1, table default)
  Advertised to update-groups:
     1
  Refresh Epoch 1
  Local
    0.0.0.0 from 0.0.0.0 (10.0.15.241)
      Origin IGP, metric 0, localpref 100, weight 32768, valid...
```

- The "show" command indicates the prefix is handled by update-group #1

Cisco's update-groups (2)

- The update group itself lists all the peers which get the same (identical) update:

```
Router1#sh ip bgp update-group 1
BGP version 4 update-group 1, internal, Address Family: IPv4 Unicast
  BGP Update version : 16/0, messages 0
  Topology: global, highest version: 16, tail marker: 16
  Format state: Current working (OK, last not in list)
                Refresh blocked (not in list, last not in list)
  Update messages formatted 11, replicated 13, current 0, refresh 0, limit 1000
  Number of NLRIs in the update sent: max 2, min 0
  Minimum time between advertisement runs is 0 seconds
  Has 13 members:
   10.0.15.242      10.0.15.243      10.0.15.244      10.0.15.245
   10.0.15.246      10.0.15.247      10.0.15.248      10.0.15.249
   10.0.15.250      10.0.15.251      10.0.15.252      10.0.15.253
   10.0.15.254
```

- And this group has 13 members

Peer Groups

- Always configure peer-groups for IBGP
  - Even if there are only a few IBGP peers
  - Easier to scale network in the future
  - Makes configuration easier to read
- Consider using peer-groups for EBGP
  - Especially useful for multiple BGP customers using same AS (RFC2270)
  - Also useful at Exchange Points:
    - Where network operator policy is generally the same to each peer
    - For Route Server where all peers receive the same routing updates

Juniper BGP groups

- JunOS has very similar configuration concept
  - Simply known as bgp groups

```
protocols {
    bgp {
        group ibgp {
            type internal;
            local-address 10.0.15.241;
            family inet {
                unicast;
            }
            export export-ibgp;
            peer-as 10;
            neighbor 10.0.15.242 {
                description "Router 2";
            }
            neighbor 10.0.15.243 {
                description "Router 3";
            }
    ...etc...
        }
    }
}
```

Dynamic Reconfiguration

Non-destructive policy changes

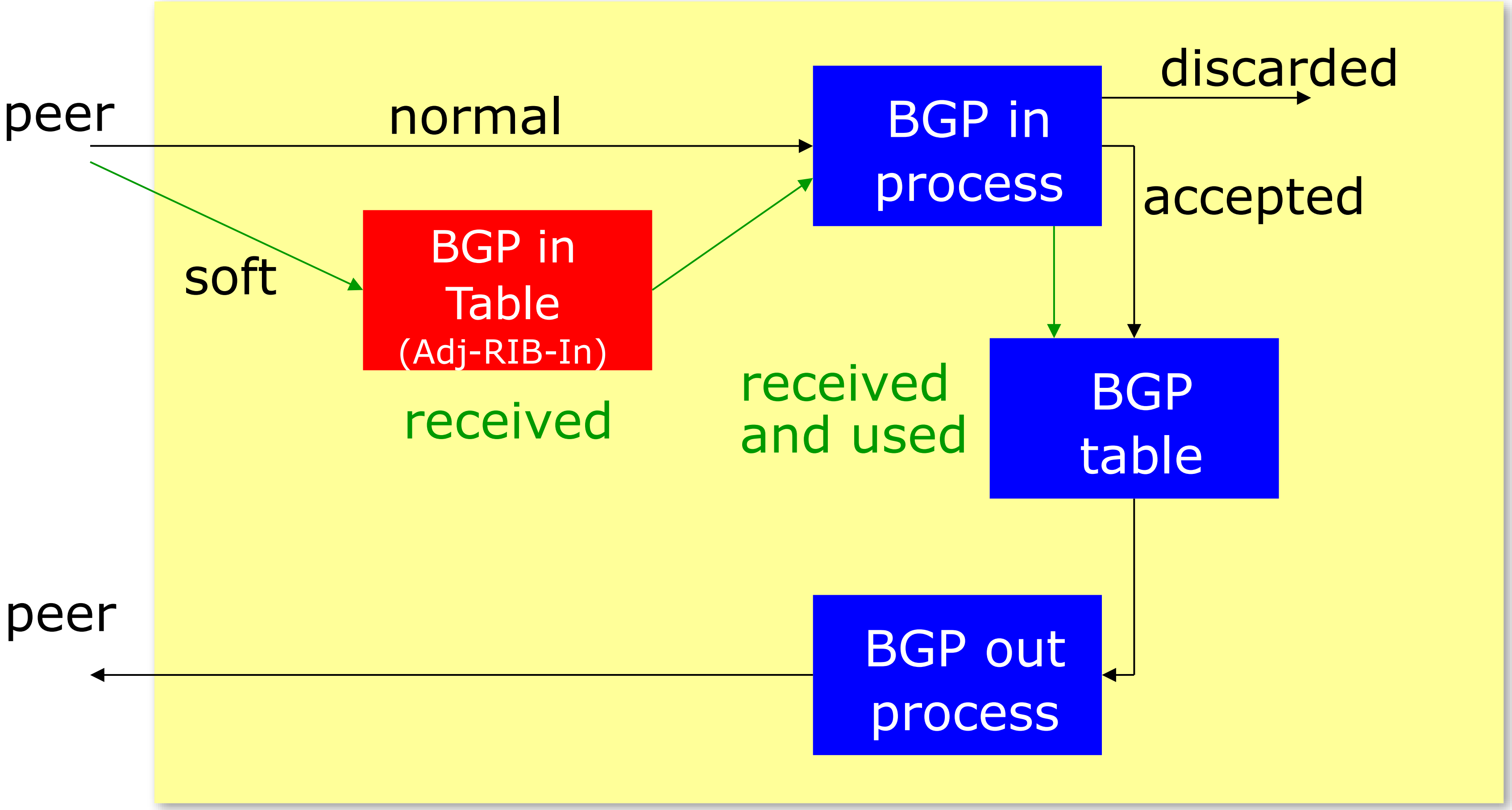
Route Refresh: History

- Historically, routers only stored prefixes which were accepted by incoming policy
  - Those rejected by policy were discarded
  - No storage of discard prefixes
- If a change of incoming policy was required:
  - The EBGP session had to be shutdown, and then brought up again
  - Destructive change: EBGP session down means lost connectivity to that peer, and potentially the rest of the Internet (outage of many minutes!)
- Changes in BGP policy usually had to be carried out during published scheduled maintenance timeslots
  - To minimise impact on end-users

Route Refresh: Step One

- First step at solving this problem was by Cisco with the "soft reconfiguration" concept
  - Router keeps a record of all prefixes received before any policy applied (known as Adj-RIB-In)
  - Needed extra memory (highly problematic in early routers and modern routers with limited memory)
    - Full BGP table with policy change could require double the control plane memory for BGP
  - Policy changes applied to the stored received prefixes
  - No shutdown and restart of the BGP session needed when implementing policy changes

Cisco's Soft Reconfiguration



Route Refresh: Step Two

- Second step at solving this problem was the introduction of “route refresh”
  - A BGP Capability: RFC2918
  - Peering remains active
  - Impacts only those prefixes affected by the policy change
  - No configuration needed
    - Automatically negotiated at peer establishment
    - No extra memory needed (no need for Adj-Rib-In)
  - Tell peer to resend full BGP announcement

```
clear ip bgp x.x.x.x [soft] in
```

  - Resend full BGP announcement to peer

```
clear ip bgp x.x.x.x [soft] out
```

Route Refresh

- Use Route Refresh capability, not hard reset
  - Supported on virtually all BGP implementations
  - Find out from “show ip bgp neighbor”
  - Non-disruptive, “Good For the Internet”

- Only hard-reset a BGP peering as a last resort

Consider the impact of a hard-reset of BGP to be equivalent to a router reboot

Route Refresh: Route Origin Validation

- Route Origin Validation means checking if the prefix received has a valid ROA
  - Route Origination Authorisation – digital object indicating the origin AS for the prefix (and subnet size) using RPKI
  - Valid ROA means that the prefix (and subnet) is being originated from the correct origin AS
  - See the “BGP Origin Validation” presentation for more in-depth content
- Routers implementing ROV apply the validation results via the existing policy language & process
  - Valid – allow; Invalid – drop; NotFound – allow (at lower preference?)
- Problem: how is incoming policy applied on routers today?

Route Refresh: Route Origin Validation

- Routers which maintain the Adj-RIB-In:
  - Apply the ROV policy to the stored received BGP table
  - Updates are applied "automatically" to the BGP table and therefore the FIB
  - No impact on any BGP peers (Route Refresh not needed)

Route Refresh: Route Origin Validation

- Routers which do NOT maintain the Adj-RIB-In:
  - Apply the ROV policy by sending a Route Refresh to peers
  - When there are a large number of ROAs (May 2026 sees around 661k IPv4 and 191k IPv6), and frequent changes or updates of ROAs:
    - Routers are sending frequent Route Refresh requests to peers (typically every few minutes)
    - Peers are being “bombarded” by Route Refresh requests: significant resource burden when they send the full or a large portion of the BGP table
    - Severe control plane CPU impact on the peer router (effectively a Denial of Service on the peer router)
  - As more and more ROAs are created and altered globally, this problem becomes significantly more serious!

Route Refresh: Route Origin Validation

- JunOS implements Adj-RIB-In by default
  - ROA updates do not cause a problem when operating ROV

- Cisco does not implement Adj-RIB-In by default:
  - Applies to all of Cisco IOS/IOS-XE/IOS-XR apart from the most recent releases
  - MUST turn on soft-reconfiguration if running ROV on the router
  - Soft-reconfiguration is similar concept to Adj-RIB-In

Enabling Cisco's Soft Reconfiguration

```
router bgp 64510
 address-family ipv4
  neighbor 100.64.1.1 remote-as 64511
  neighbor 100.64.1.1 route-map infilter in
  neighbor 100.64.1.1 soft-reconfiguration inbound
```

- When the policy needs to be changed:

```
clear ip bgp 100.64.1.1 soft in
```

- Note:
  - When "soft-reconfiguration" is enabled, there is no access to the route-refresh capability CLI
  - `clear ip bgp 100.64.1.1 in` also does a soft refresh

Using Cisco's Soft-Reconfiguration

- Strongly recommended when deploying Route Origin Validation
- Operators will also use soft-reconfiguration when trouble-shooting EBGP peer problems
  - Soft reconfiguration enabled on an EBGP session means that the operator can see which prefixes were sent by a neighbour before any policy is applied
  - This helps saves arguments between operators about whose BGP filters may have configuration errors!

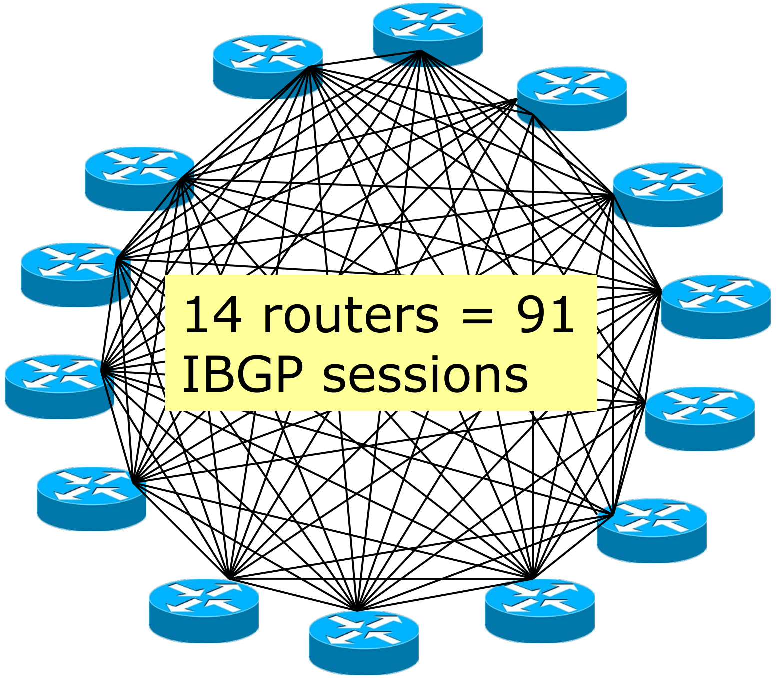
Route Reflectors

Scaling the IBGP mesh

Scaling the IBGP mesh

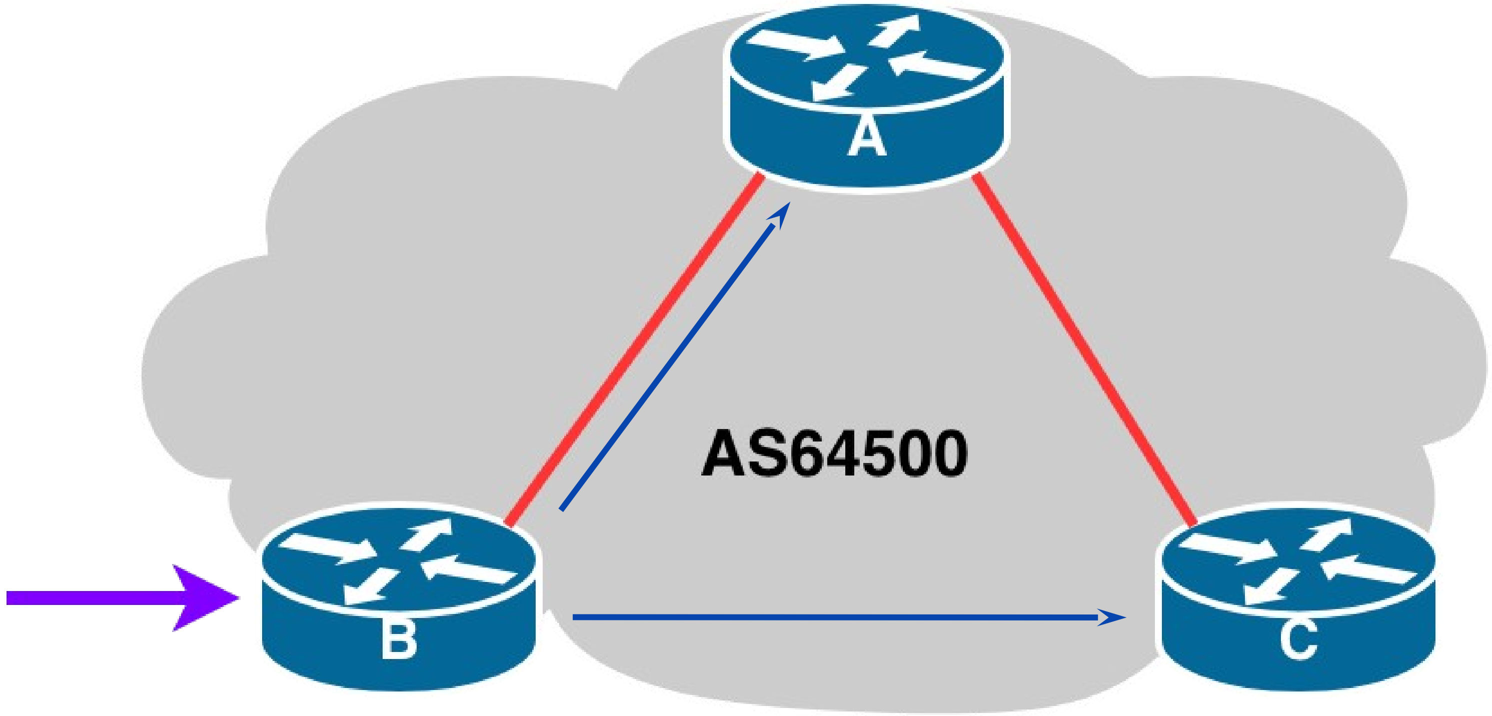
Avoid ½n(n-1) IBGP mesh

n=1000 ⇒ nearly half a million IBGP sessions!

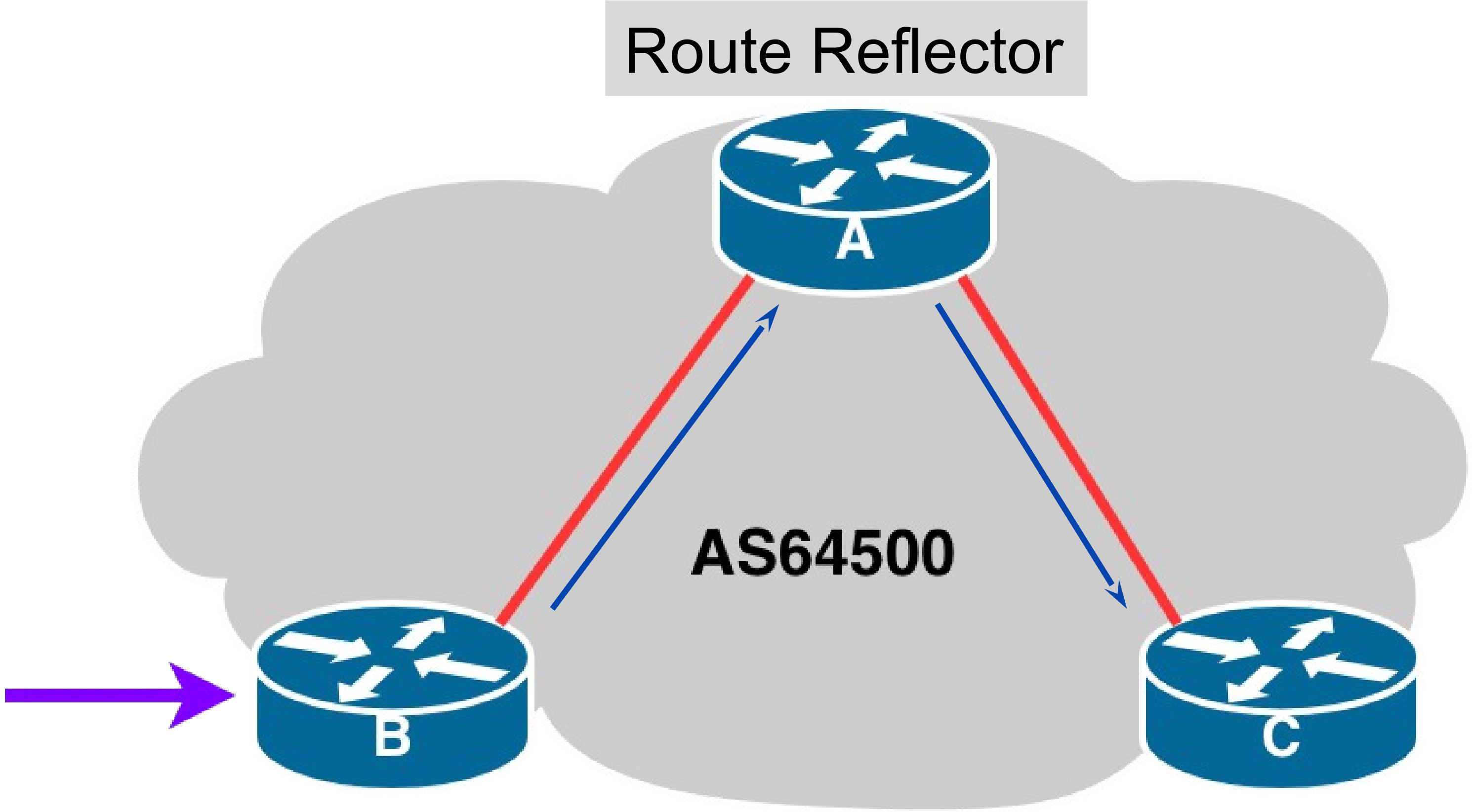


- Two solutions
  - Route reflector: simpler to deploy and run
  - BGP Confederation: more complex, has corner case advantages

Route Reflector: Principle

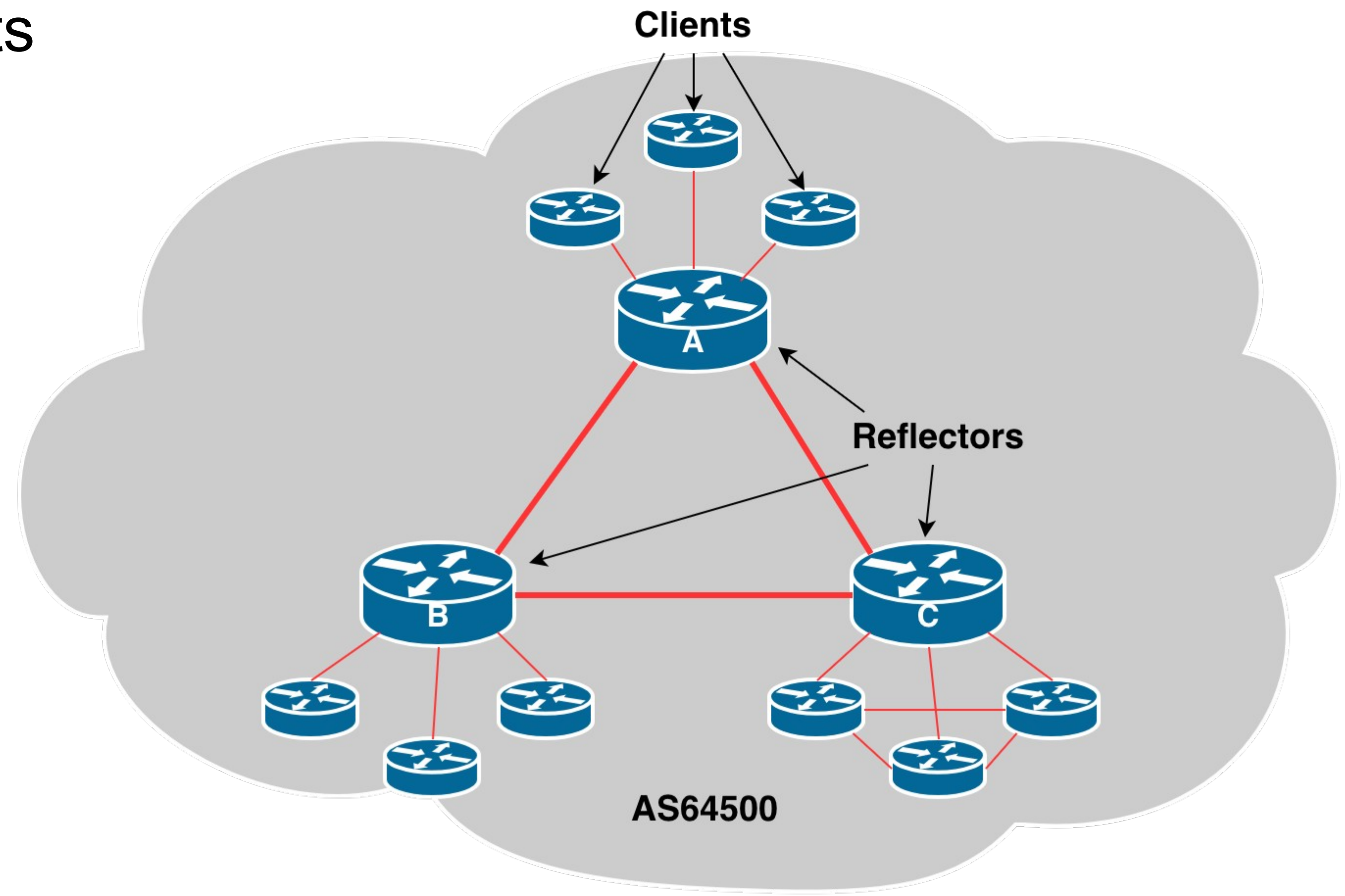


Route Reflector: Principle



Route Reflector: Rules

- Reflector receives path from clients and non-clients
- Selects best path
- If best path is from client, reflect to other clients and non-clients
- If best path is from non-client, reflect to clients only
- Non-meshed clients
- Described in RFC4456



Route Reflector: Topology

- Divide the backbone into multiple clusters
- Provision at least one Route Reflector (RR) and few clients per cluster
- Route reflectors are fully meshed
- Clients in a cluster could be fully meshed
- Single IGP still carries next-hop and any local routes

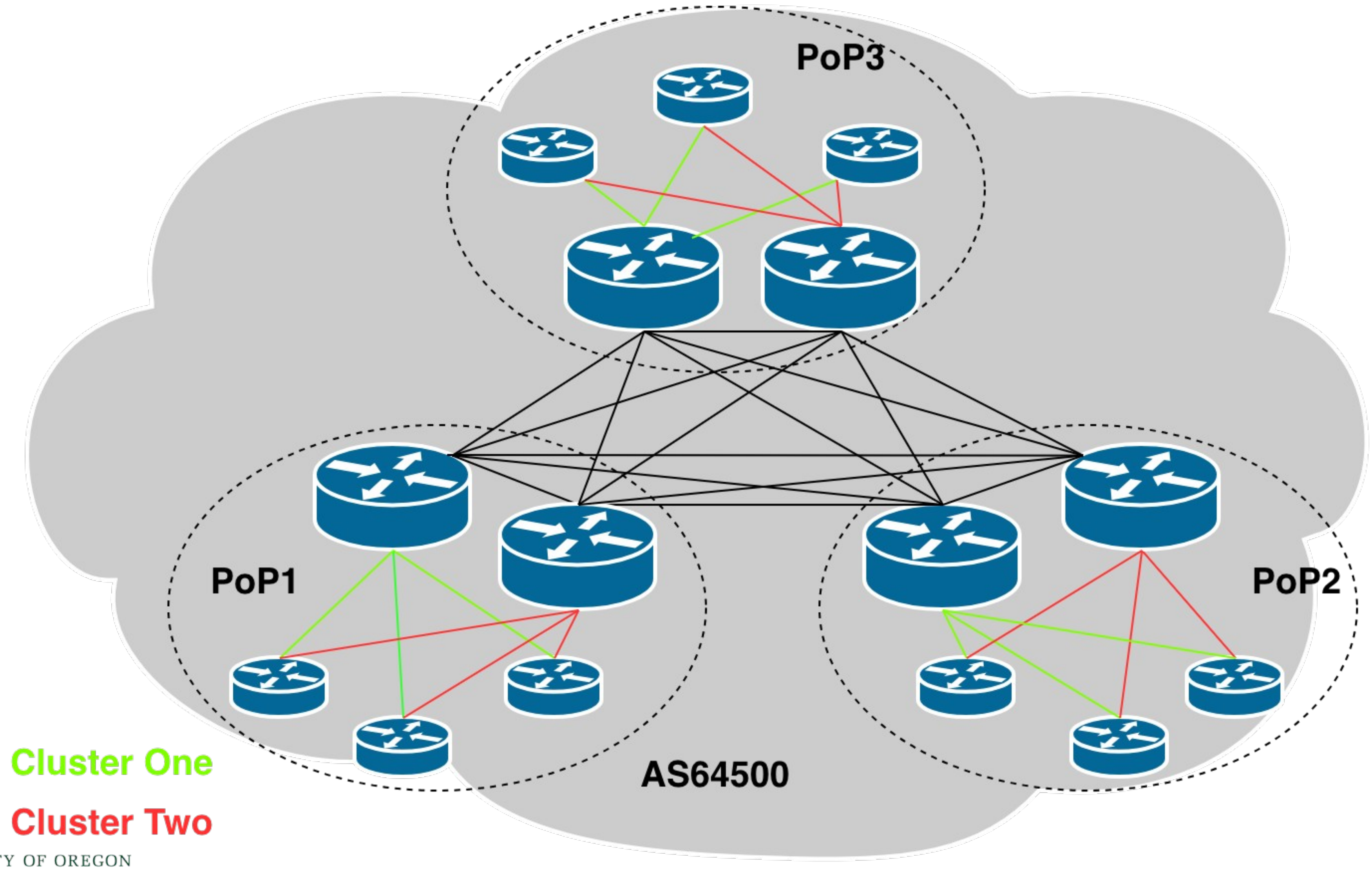
Route Reflector: Loop Avoidance

- `Originator_ID` attribute
  - Carries the RID of the originator of the route in the local AS (created by the RR)
- `Cluster_list` attribute
  - The local `cluster-id` is added when the update is sent by the RR
  - `Cluster-id` is the same as the `router-id` by default (usually the address of loopback interface)
  - Never use `bgp cluster-id x.x.x.x`
    - Only exception is if the two route reflectors are physically/directly connected
      - (And even then, consideration needed of the potential failure scenarios)

Route Reflector: Redundancy

- Multiple RRs can be configured in the same cluster – not advised!
  - All RRs in the cluster must have the same cluster-id (otherwise it is a different cluster)
- A router may be a client of RRs in different clusters
  - Common today in service provider networks to overlay two clusters – redundancy achieved that way
  - → Each client has two RRs = redundancy

Route Reflector: Redundancy



Route Reflector: Benefits

- Solves IBGP mesh problem
- Packet forwarding is not affected
- Normal BGP speakers co-exist
- Multiple reflectors for redundancy
- Easy migration
- Multiple levels of route reflectors

Route Reflector: Deployment

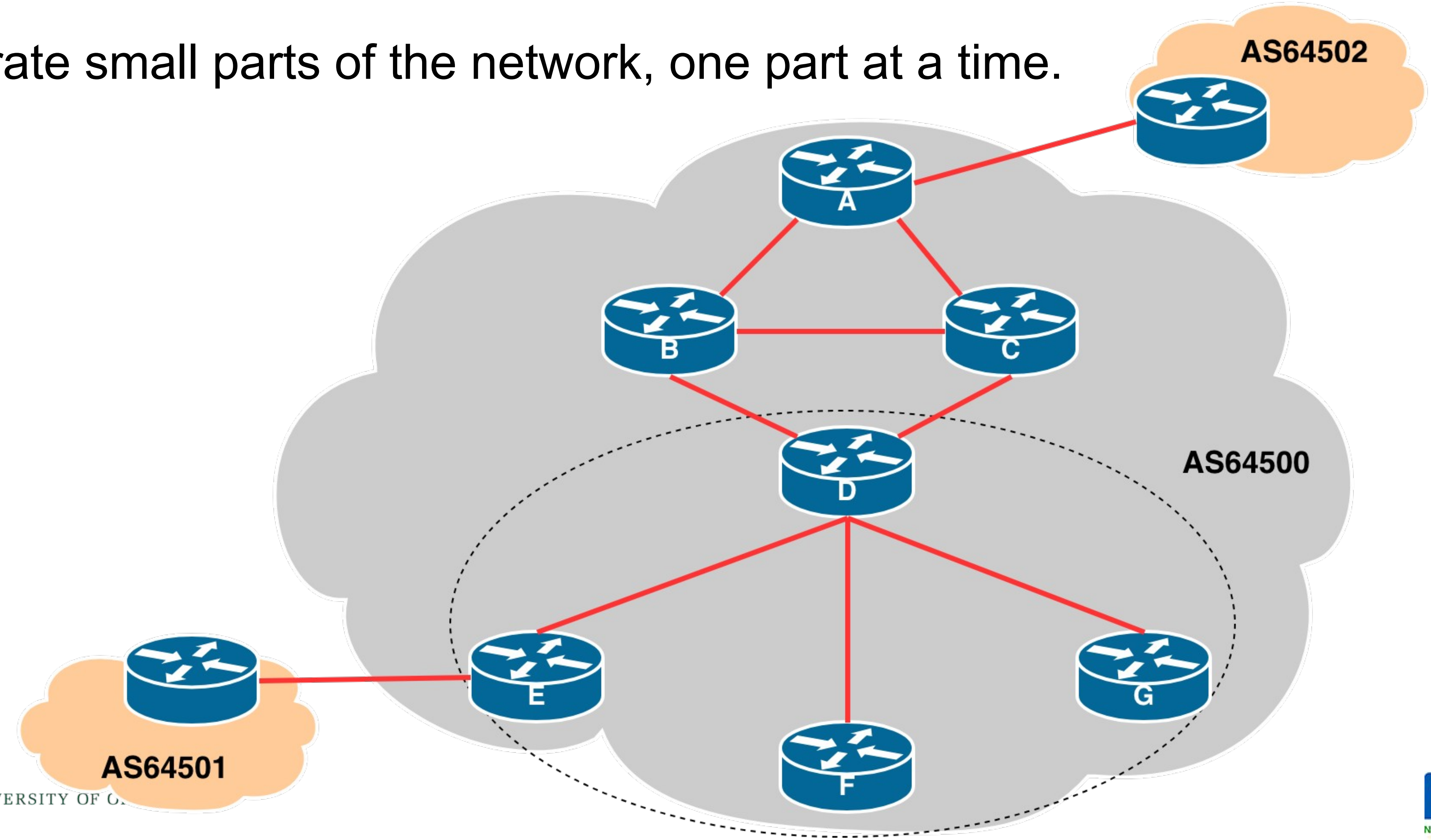
- Where to place the route reflectors?
  - Always follow the physical topology!
  - This will guarantee that the packet forwarding won't be affected
- Typical Service Provider network:
  - PoP has two core routers
  - Core routers are RR for the PoP
  - Two overlaid clusters

Route Reflector: Migration

- Typical service provider network:
  - Core routers have fully meshed IBGP
  - Create further hierarchy if core mesh too big
    - Split backbone into regions
- Configure one cluster pair at a time
  - Eliminate redundant IBGP sessions
  - Place maximum one RR per cluster
  - Easy migration, multiple levels

Route Reflector: Migration

- Migrate small parts of the network, one part at a time.



Route Reflector: Cisco IOS Configuration

- Router D configuration:

```
router bgp 64500
 address-family ipv4
 ...
  neighbor 100.64.3.4 remote-as 64500
  neighbor 100.64.3.4 route-reflector-client
  neighbor 100.64.3.5 remote-as 64500
  neighbor 100.64.3.5 route-reflector-client
  neighbor 100.64.3.6 remote-as 64500
  neighbor 100.64.3.6 route-reflector-client
 ...
```

BGP Scaling Techniques

- These two standards-based techniques must be designed in from the beginning for all network operator infrastructure
  - Route Refresh
  - Route Reflectors

BGP Confederations

Confederations

- Divide the AS into sub-AS
  - EBGP between sub-AS, but some IBGP information is kept
    - Preserve NEXT_HOP across the sub-AS (IGP carries this information)
    - Preserve LOCAL_PREF and MED
- Usually a single IGP
- Described in RFC5065

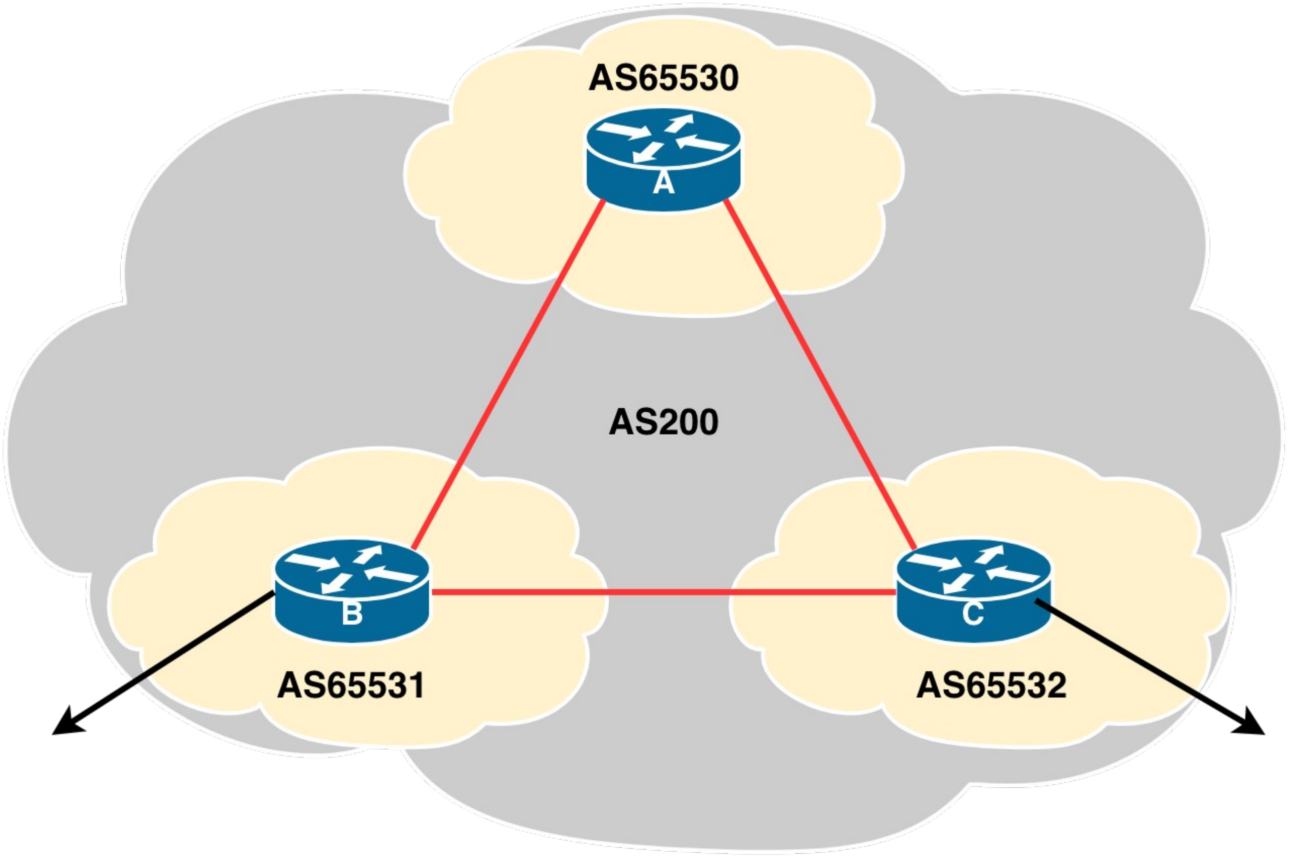
Confederations

- Visible to outside world as single AS – "Confederation Identifier"
  - Each sub-AS uses a number from the private space (64512-65534)
- IBGP speakers in sub-AS are fully meshed
  - The total number of neighbors is reduced by limiting the full mesh requirement to only the peers in the sub-AS
  - Can also use Route-Reflector within sub-AS

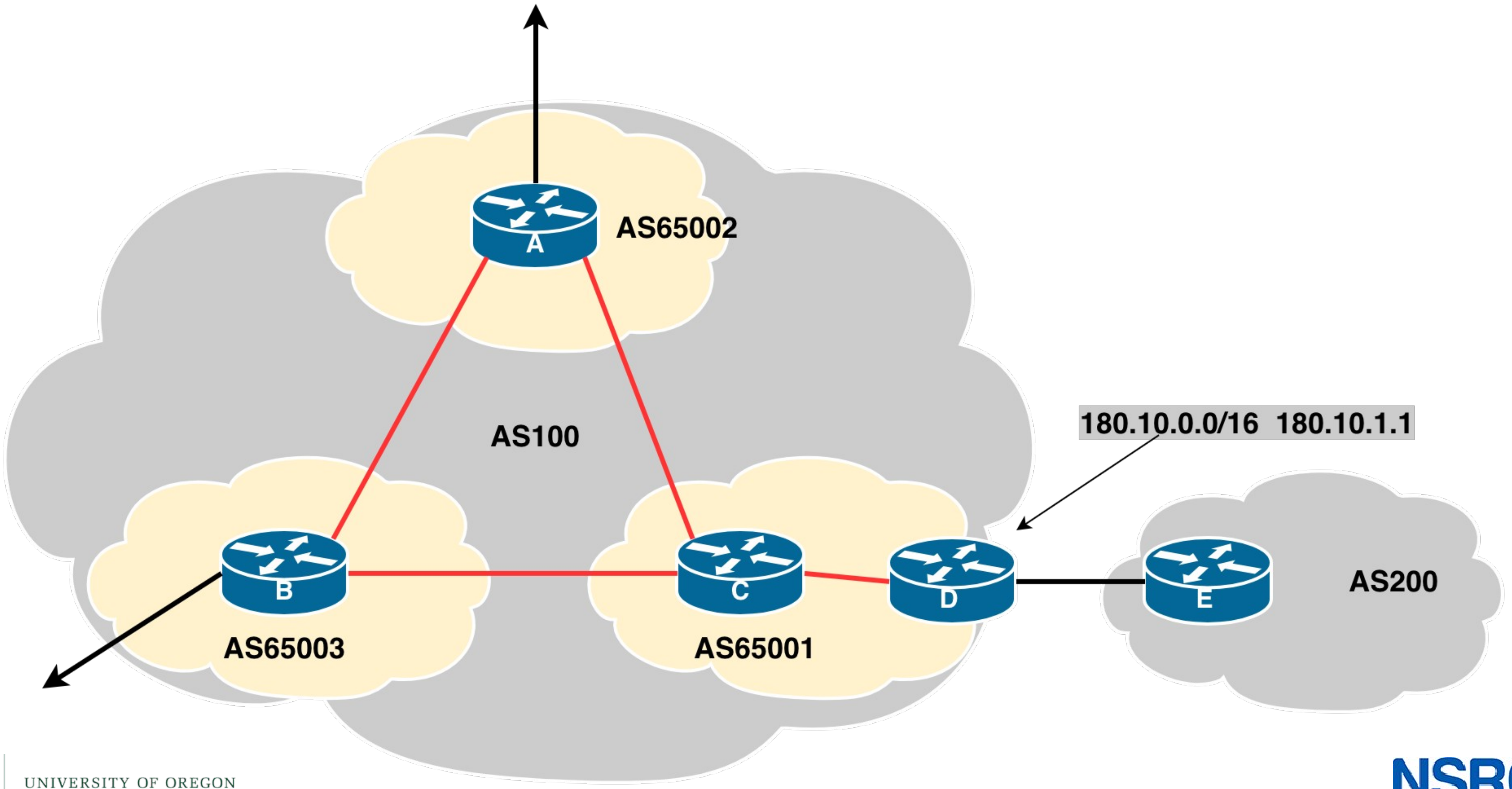
Confederations

Configuration (Router C):

```
router bgp 65532
 bgp confederation identifier 200
 bgp confederation peers 65530 65531
 neighbor 141.153.12.1 remote-as 65530
 neighbor 141.153.17.2 remote-as 65531
```



Confederations: Next Hop



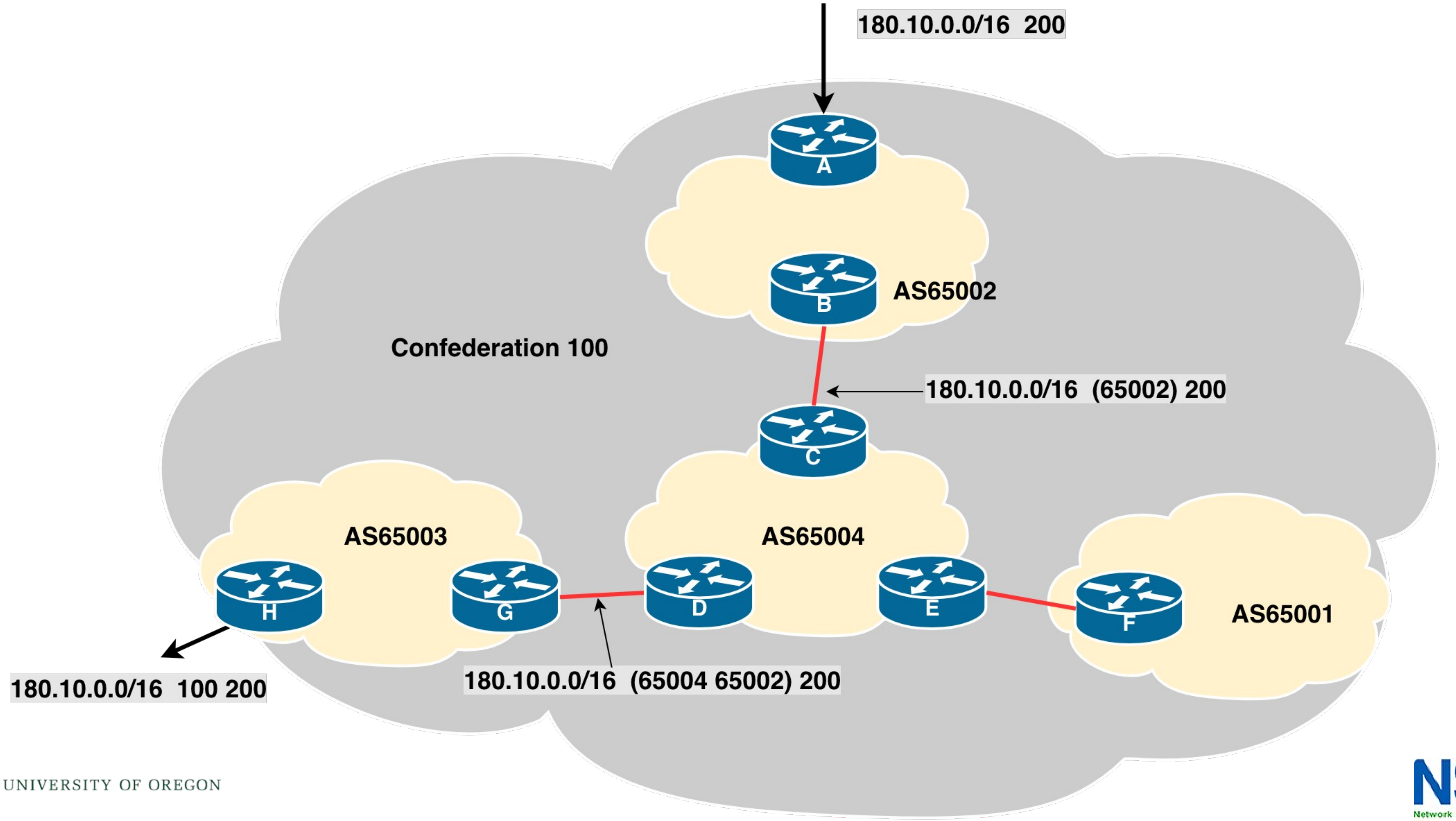
Confederations: Principle

- Local preference and MED influence path selection
- Preserve local preference and MED across sub-AS boundary
- Sub-AS EBGP path administrative distance

Confederations: Loop Avoidance

- Sub-AS traversed are carried as part of AS-path
- AS-sequence and AS path length
- Confederation boundary
- AS-sequence should be skipped during MED comparison

Confederations: AS-Sequence



Route Propagation Decisions

- Same as with “normal” BGP:
  - From peer in same sub-AS → only to external peers
  - From external peers → to all neighbors
- “External peers” refers to
  - Peers outside the confederation
  - Peers in a different sub-AS
    - Preserve LOCAL_PREF, MED and NEXT_HOP

Confederations (cont.)

- Example (cont.):

```
BGP table version is 78, local router ID is 141.153.17.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

Network          Next Hop      Metric LocPrf Weight Path
*> 10.0.0.0      141.153.14.3    0     100       0    (65531) 1 i
*> 141.153.0.0 141.153.30.2    0     100       0    (65530) i
*> 144.10.0.0   141.153.12.1    0     100       0    (65530) i
*> 199.10.10.0 141.153.29.2    0     100       0    (65530) 1 i
```

More points about confederations

- Can ease “absorbing” other service providers into your provider
  - e.g., if one provider buys another
  - (can use local-as feature to do a similar thing)
- You can use route-reflectors with confederation sub-AS to reduce the sub-AS IBGP mesh

Confederations: Benefits

- Solves IBGP mesh problem
- Packet forwarding not affected
- Can be used with route reflectors
- Policies could be applied to route traffic between sub-Ases if required

Confederations: Caveats

- Minimal number of sub-AS
- Sub-AS hierarchy
- Minimal inter-connectivity between sub-ASes
- Path diversity
- Difficult migration
  - BGP reconfigured into sub-AS
  - Must be applied across the network

RRs or Confederations ?

	Internet Connectivity	Multi-Level Hierarchy	Policy Control	Scalability	Migration Complexity
Confederations	Anywhere in the network	Yes	Yes	Medium	Medium to High
Route Reflectors	Anywhere in the network	Yes	Yes	Very High	Very Low

New network operators deploy Route Reflectors from Day One

BGP Communities

Scaling BGP Policies

Multihoming and Communities

- The BGP community attribute is a very powerful tool for assisting and scaling BGP Policies and BGP Multihoming
- Most major Network Operators make extensive use of BGP communities:
  - Internal policies (IBGP)
  - Inter-provider relationships (MED replacement)
  - Customer traffic engineering

Communities for IBGP

- Network Operators tag prefixes learned from their BGP and static customers with communities
  - To identify services the customer may have purchased
  - To identify prefixes which are part of the Provider’s PA space
  - To identify PI customer addresses
  - To control prefix distribution in IBGP
  - To control prefix announcements to customers and upstreams
  - (amongst several other reasons)

Example communities for IBGP

Community Value	Description
X:1000	Aggregates
X:1001	Subprefixes of AS X aggregates
X:1005	Static (non-BGP) customer Provider Independent addresses
X:2000	BGP customers who get Transit
X:2100	BGP customers announced to Private Peers
X:2200	BGP customers announced at IXPs
X:2500	BGP customers announced to other BGP customers
X:3100	Routes received from Private Peers
X:3200	Routes received from IXP peers

Network Operator BGP Communities

- There are no recommended Network Operator BGP communities apart from
  - RFC1998
  - The well-known communities
    - https://www.iana.org/assignments/bgp-well-known-communities
- Efforts have been made to document from time to time
  - Collection of Network Operator communities at https://www.onesc.net/communities
  - NANOG Tutorial:
    - https://www.nanog.org/meetings/nanog40/presentations/BGPcommunities.pdf
- Network Operator policy is usually published
  - On the Operator’s website
  - Referenced in the AS Object in the IRR

Communities for EBGP

Community Value	Action	Description
X:80	`set local-preference 80`	Backup path
X:120	`set local-preference 120`	Primary path (over-ride BGP path selection default)
X:1	`set as-path prepend X`	Single prepend when announced to X's upstreams
X:2	`set as-path prepend X X`	Double prepend when announced to X's upstreams
X:3	`set as-path prepend X X X`	Triple prepend when announced to X's upstreams
X:666	`set ip next-hop 192.0.2.1`	Blackhole route – very useful for DoS attack mitigation (RFC7999)

Sample Router Configuration (1)

```
router bgp 100
 address-family ipv4
  neighbor 100.66.32.2 remote-as 130
  neighbor 100.66.32.2 route-map customer-policy-in in
  neighbor 100.66.32.2 activate
  neighbor 100.65.8.9 remote-as 200
  neighbor 100.65.8.9 route-map upstream-out out
  neighbor 100.65.8.9 activate
!
ip community-list standard prepend-1 permit 100:1
ip community-list standard prepend-2 permit 100:2
ip community-list standard prepend-3 permit 100:3
ip community-list standard lp-80     permit 100:80
ip community-list standard lp-120    permit 100:120
ip community-list standard RTBH      permit 100:666
!
ip route 192.0.2.1 255.255.255.255 null0
```

Customer BGP

Upstream BGP

Black hole route
(on all routers)

Sample Router Configuration (2)

```
route-map customer-policy-in permit 10
 match community lp-80
 set local-preference 80
!
route-map customer-policy-in permit 20
 match community lp-120
 set local-preference 120
!
route-map customer-policy-in permit 30
 match community RTBH
 set ip next-hop 192.0.2.1
!
route-map customer-policy-in permit 40

...etc...
```

Sample Router Configuration (3)

```
route-map upstream-out permit 10
 match community prepend-1
 set as-path prepend 100
!
route-map upstream-out permit 20
 match community prepend-2
 set as-path prepend 100 100
!
route-map upstream-out permit 30
 match community prepend-3
 set as-path prepend 100 100 100
!
route-map upstream-out permit 40

...etc...
```

BGP Communities

- There are no “standard communities” representing what Network Operators do
- Best practices today consider that Network Operators should use BGP communities extensively for:
  - Scaling IBGP
  - Multihoming support of traffic engineering
- Look in the Network Operator AS Object in the IRR or on their website for documented community support

Route Flap Damping

Network Stability for the 1990s

Network Instability for the 21st Century!

Route Flap Damping

- For many years, Route Flap Damping was a strongly recommended practice
- Now it is strongly discouraged as it causes far greater network instability than it cures
- But first, the theory…

Route Flap Damping

- Route flap
  - Going up and down of path or change in attribute
    - BGP WITHDRAW followed by UPDATE = 1 flap
    - EBGP neighbour going down/up is NOT a flap
  - Ripples through the entire Internet
  - Wastes CPU
- Damping aims to reduce scope of route flap propagation

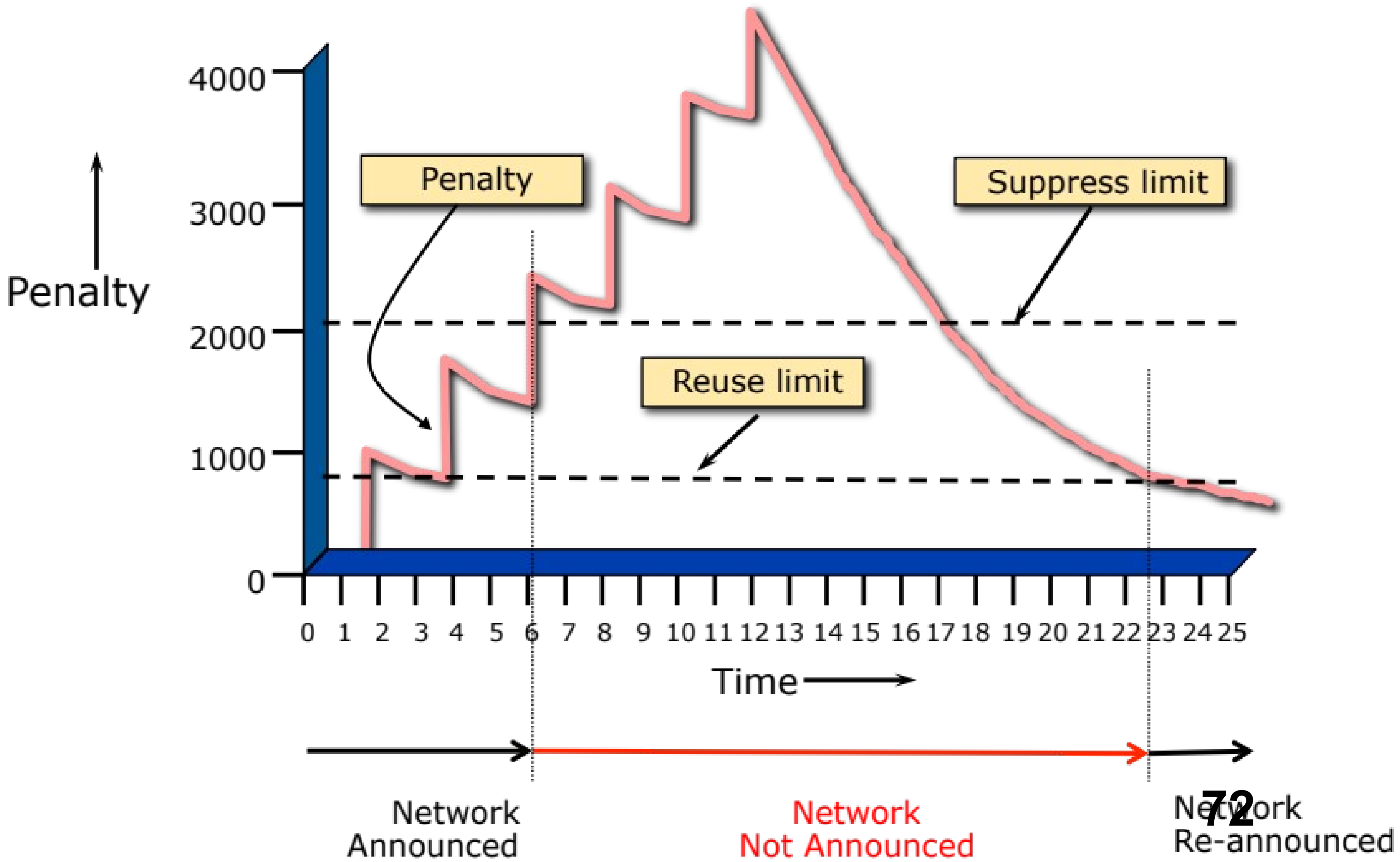
Route Flap Damping (continued)

- Requirements
  - Fast convergence for normal route changes
  - History predicts future behaviour
  - Suppress oscillating routes
  - Advertise stable routes
- Implementation described in RFC 2439

Operation

- Add penalty (1000) for each flap
  - Change in attribute gets penalty of 500
- Exponentially decay penalty
  - Half life determines decay rate
- Penalty above suppress-limit
  - Do not advertise route to BGP peers
- Penalty decayed below reuse-limit
  - Re-advertise route to BGP peers
  - Penalty reset to zero when it is half of reuse-limit

Operation


Penalty
4000
3000
2000
1000
0
Penalty
Suppress limit
Reuse limit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
Time
Network Announced
Network Not Announced
Network Re-announced

Operation

- Only applied to inbound announcements from EBGP peers
- Alternate paths still usable
- Controlled by:
  - Half-life (default 15 minutes)
  - reuse-limit (default 750)
  - suppress-limit (default 2000)
  - maximum suppress time (default 60 minutes)

Configuration

- Fixed damping

```
router bgp 100
 bgp dampening [<half-life> <reuse-value> <suppress-penalty> <max suppress time>]
```

- Selective and variable damping

```
bgp dampening [route-map <name>]
route-map <name> permit 10
 match ip address prefix-list FLAP-LIST
 set dampening [<half-life> <reuse-value> <suppress-penalty> <max suppress time>]
ip prefix-list FLAP-LIST permit 192.0.2.0/24 le 32
```

Operation

- Care required when setting parameters
- Penalty must be less than reuse-limit at the maximum suppress time
- Maximum suppress time and half life must allow penalty to be larger than suppress limit

Configuration

- Examples –
  - bgp dampening 15 500 2500 30
    - reuse-limit of 500 means maximum possible penalty is 2000 – no prefixes suppressed as penalty cannot exceed suppress-limit
- Examples –
  - bgp dampening 15 750 3000 45
    - reuse-limit of 750 means maximum possible penalty is 6000 – suppress limit is easily reached

Maths!

- Maximum value of penalty is

$$\text{max-penalty} = \text{reuse-limit} \times 2^{\left(\frac{\text{max-suppress-time}}{\text{half-life}}\right)}$$

- Always make sure that suppress-limit is LESS than max-penalty otherwise there will be no route damping

Route Flap Damping History

- First implementations on the Internet by 1995
- Vendor defaults too severe
  - RIPE Routing Working Group recommendations in ripe-178, ripe-210, and ripe-229
  - www.ripe.net/ripe/docs
  - But many network operators simply switched on the vendors' default values without thinking

Serious Problems:

- “Route Flap Damping Exacerbates Internet Routing Convergence”
  - Zhuoqing Morley Mao, Ramesh Govindan, George Varghese & Randy H. Katz, August 2002
    http://conferences.sigcomm.org/sigcomm/2002/papers/routedampening.pdf
- “What is the sound of one route flapping?”
  - Tim Griffin, June 2002
    - For example: https://ripe43.ripe.net/presentations/ripe43-routing-flap.pdf
- Various work on routing convergence by Craig Labovitz and Abha Ahuja a few years prior to the above research work
- “Happy Packets”
  - Closely related work by Randy Bush et al

Problem 1:

- One path flaps:
  - BGP speakers pick next best path, announce to all peers, flap counter incremented
  - Those peers see change in best path, flap counter incremented
  - After a few hops, peers see multiple changes simply caused by a single flap → prefix is suppressed

Problem 2:

- Different BGP implementations have different transit time for prefixes
  - Some hold onto prefix for some time before advertising
  - Others advertise immediately
- Race to the finish line causes appearance of flapping, caused by a simple announcement or path change → prefix is suppressed

Solution:

- Misconfigured Route Flap Damping will seriously impact access to:
  - Your network *and*
  - The Internet
- More background contained in RIPE Routing Working Group document:
  - www.ripe.net/ripe/docs/ripe-378
- Recommendations now in:
  - www.rfc-editor.org/rfc/rfc7196.txt and www.ripe.net/ripe/docs/ripe-580

Questions?