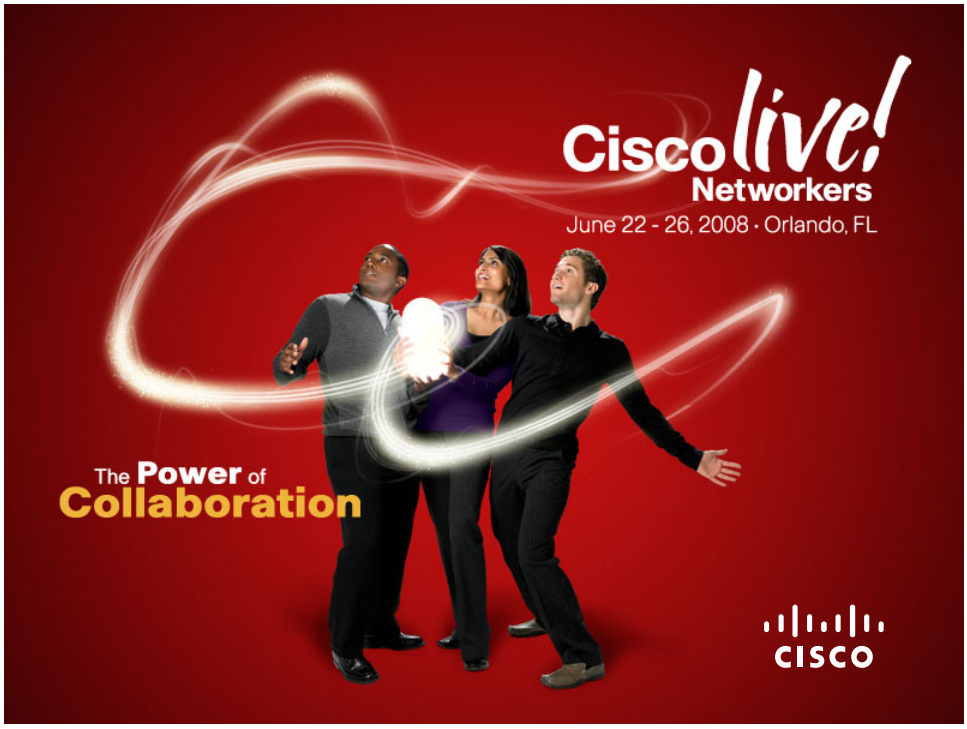
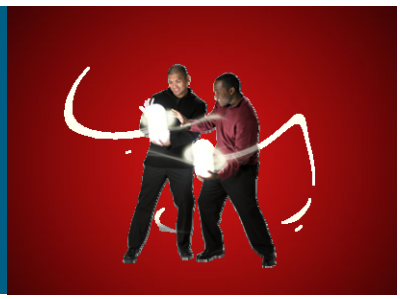


CISCO

Cisco live! Networkers

Routing Operation in Cisco Routers

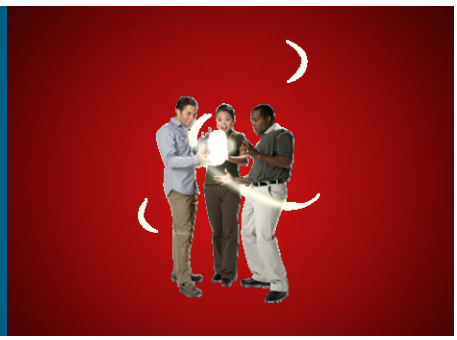


BRKRST-2350

Routing Operation in Cisco Routers

- The Routing Table (RIB)
- Route Redistribution
- Filtering Routes
- Load Sharing

The Routing Table

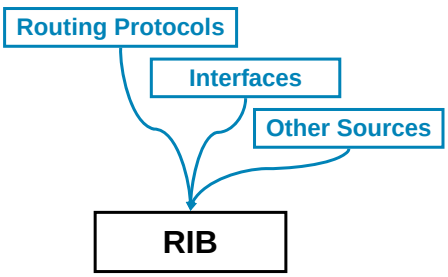


The Routing Table

- Basic Structure
- Gateway of Last Resort
- Interface Down Events
- IP Event Dampening
- Route Selection
- Backup Routes
- Static Routes
- Discard Routes

Basic Structure

- The Routing Information Base, or RIB
- Routing Protocols
  - Install routes into the RIB
  - Static routing is a routing protocol
- Interfaces
  - Install routes into the RIB
- Other Sources
  - Install routes into the RIB



Basic Structure

- Routing Protocols

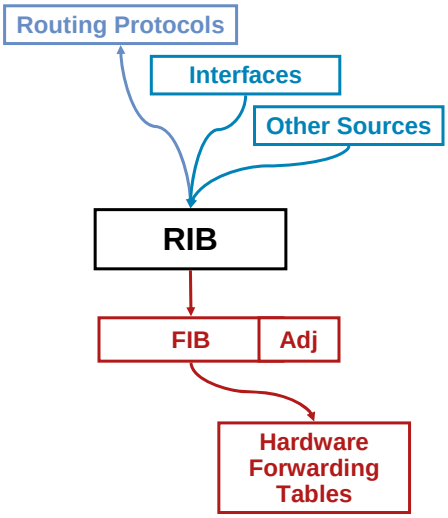
  Pull routes from the RIB for redistribution

- Cisco Express Forwarding (CEF)

  CEF maintains the FIB, Forwarding Information Base, and the Adjacency tables

  A copy of the RIB is sent down to the FIB

  A copy of the RIB is sent down to the hardware forwarding component



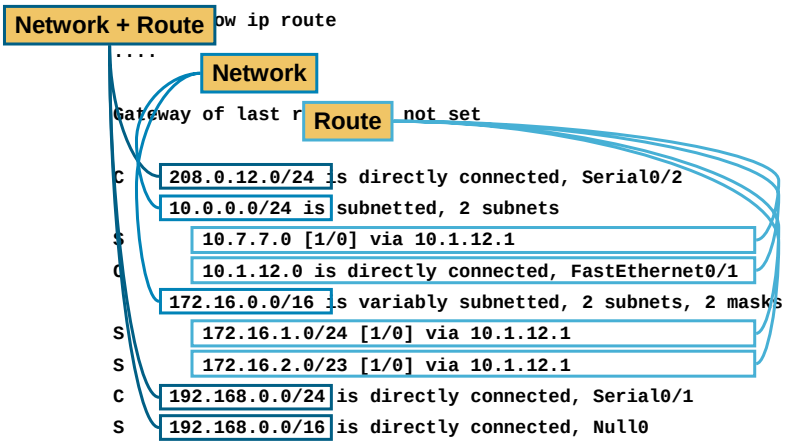
Basic Structure

```
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

C    208.0.12.0/24 is directly connected, Serial0/2
     10.0.0.0/24 is subnetted, 2 subnets
S       10.7.7.0 [1/0] via 10.1.12.1
C       10.1.12.0 is directly connected, FastEthernet0/1
     172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
S       172.16.1.0/24 [1/0] via 10.1.12.1
S       172.16.2.0/23 [1/0] via 10.1.12.1
C    192.168.0.0/24 is directly connected, Serial0/1
S    192.168.0.0/16 is directly connected, Null0
```

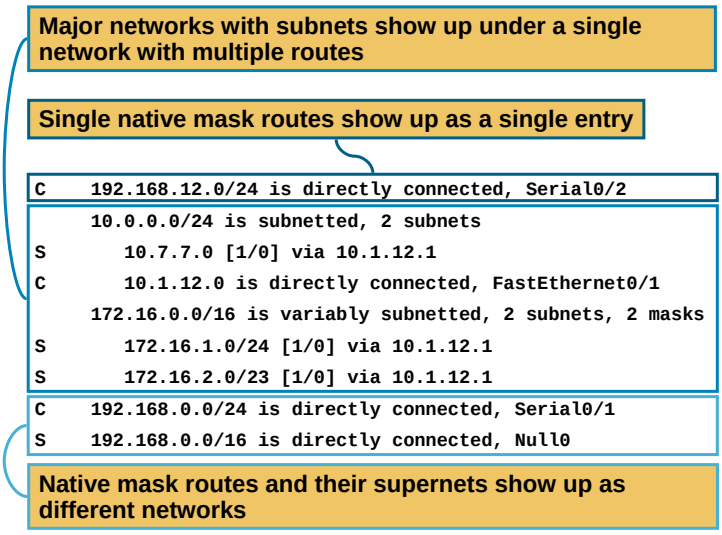
Basic Structure (Notes)

- Each code relates to the protocol or process that installed the route into the routing table
- “Static” and “connected” are treated as routing processes by the routing table

Basic Structure



Basic Structure



Basic Structure

- The administrative distance is used to determine which route among several possible routes is installed in the routing table
  - There's more information on this later in this presentation
- The time shown is the amount of time since the route was touched
  - EIGRP recalculation of any type, including losing an alternate path, resets this timer
  - OSPF SPF run resets this timer
  - IS-IS SPF run resets this timer

Gateway of Last Resort

```
Router#show ip route
....
Gateway of last resort is not set
```

No default next hop for unknown destinations

```
....
ip default-network 192.168.1.0
ip route 192.168.1.0 255.255.255.0 10.1.12.1
....
Router#show ip route
....
Gateway of last resort is 10.1.12.1 to network 192.168.1.0
```

ip default-network configured

```
....
ip route 0.0.0.0 0.0.0.0 10.1.12.1
....
Router#show ip route
....
Gateway of last resort is 10.1.12.1 to network 0.0.0.0
```

ip route 0.0.0.0 0.0.0.0 configured

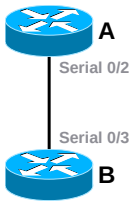
Gateway of Last Resort

- `ip default network`
  - Designed for IGRP
  - EIGRP is the only other routing protocol that will propagate a default based on the ip default network
  - Not recommended!
  - Will probably be removed from Cisco IOS® Software in the near future
- Default route
  - `ip route 0.0.0.0 0.0.0.0 <next hop|interface>`
  - Recommended for injecting a default route into the network
  - All routing protocols except IGRP will propagate a default based on the default route

Interface Down Events

- How does the RIB interact with the routing protocol when an interface fails?
- We can use debug ip routing and debug ip eigrp events to examine this interaction
- Assume the interface between A and B fails; what will we see at A?

```
interface Serial0/2
 ip address 192.168.12.10 255.255.255.0
 clockrate 1000000
!
router eigrp 100
 network 192.168.12.0
```

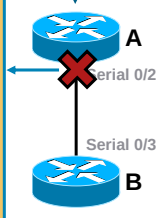



```
interface Serial0/3
 ip address 192.168.12.11 255.255.255.0
!
router eigrp 100
 network 0.0.0.0
 no eigrp log-neighbor-changes
```


Interface Down Events

```
A#show ip route
....

D    192.168.13.0/24 [90/2681856] via 192.168.12.11, Serial0/2
D    192.168.8.0/24  [90/2681856] via 192.168.12.11, Serial0/2
C    192.168.12.0/24 is directly connected, Serial0/2
....
```

```
13:08:39.618: IP-EIGRP: Callback: route_adjust Serial0/2
13:08:39.618: RT: interface Serial0/2 removed from routing table
13:08:39.622: RT: del 192.168.12.0 via 0.0.0.0, connected metric [0/0]
13:08:39.622: RT: delete network route to 192.168.12.0
13:08:39.622: IP-EIGRP: Callback: ignored connected
              AS 0 192.168.12.0/24 into: eigrp AS 100
13:08:39.622: RT: Pruning routes for Serial0/2 (3)
13:08:39.622: RT: delete route to 192.168.13.0 via 192.168.12.11,
              Serial0/2
13:08:39.622: RT: no routes to 192.168.13.0, flushing
13:08:39.626: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 100: Neighbor
192.168.12.11 (Serial0/2) is down: interface down
13:08:39.630: RT: delete route to 192.168.8.0 via 192.168.12.11,
              Serial0/2
13:08:39.630: RT: no routes to 192.168.8.0, flushing
13:08:39.634: IP-EIGRP(Default-IP-Routing-Table:100): Callback:
reload_iptable Serial0/2
13:08:41.617: %LINK-5-CHANGED: Interface Serial0/2,
              changed state to down
13:08:42.619: %LINEPROTO-5-UPDOWN: Line protocol on
              Interface Serial0/2, changed state to down
```



Interface Down Events

The RIB tells EIGRP the interface is down

192.168.12.0, connected, is removed from the RIB

192.168.13.0, learned through EIGRP, is removed from the RIB (before EIGRP takes the neighbor down)

The EIGRP neighbor goes down

192.168.8.0, learned through EIGRP, is removed from the RIB

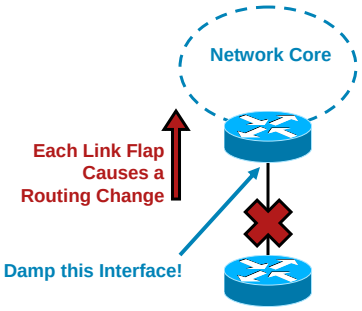
EIGRP is notified it should load any backup routes for the routes just deleted

The interface changes to down state

```
39.618: IP-EIGRP: Callback: route_adjust
        Serial0/2
39.618: RT: interface Serial0/2 removed from
        routing table
39.622: RT: del 192.168.12.0 via 0.0.0.0,
        connected metric [0/0]
39.622: RT: delete network route to
        192.168.12.0
39.622: IP-EIGRP: Callback: ignored connected
        AS 0 192.168.12.0/24 into: eigrp AS 100
39.622: RT: Pruning routes for Serial0/2 (3)
39.622: RT: delete route to 192.168.13.0 via
        192.168.12.11, Serial0/2
39.622: RT: no routes to 192.168.13.0, flushing
39.626: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 100:
        Neighbor 192.168.12.11 (Serial0/2) is
        down: interface down
39.630: RT: delete route to 192.168.8.0 via
        192.168.12.11, Serial0/2
39.630: RT: no routes to 192.168.8.0, flushing
39.634: IP-EIGRP(Default-IP-Routing-Table:100):
        Callback: reload_iptable Serial0/2
41.617: %LINK-5-CHANGED: Interface Serial0/2,
        changed state to down
42.619: %LINEPROTO-5-UPDOWN: Line protocol on
        Interface Serial0/2, changed state to
        down
```


IP Event Dampening

- A flapping link can cause major problems many hops away
- Even if you are using good network design techniques, like summarization, link flaps can still cause a major portion of your network to converge with each flap
- IP event dampening catches the problem at its source, the flapping interface



IP Event Dampening

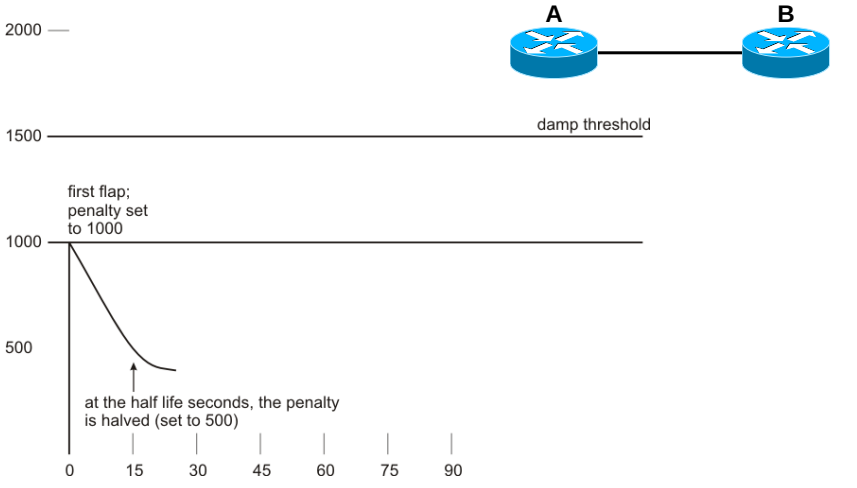
- IP event dampening is configured on a per interface basis


```
router-a(config-if)#dampening 15 1000 1500 60 1000
```

half-life-period	the penalty is decreased by half each half life period; 1 to 30 seconds
reuse-threshold	when the penalty drops below this level, damping will be removed; 1 to 20000
suppress-threshold	when the penalty reaches this level, the interface will be dampened; 1 to 20000
max-suppress-time	maximum amount of time the interface will be damped; 1 to 20000 seconds
restart-penalty	amount of penalty applied for each interface restart; 1 to 20000

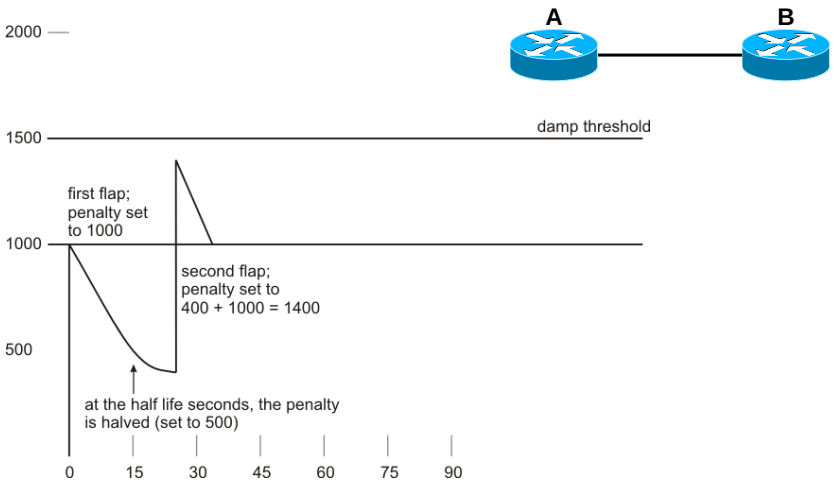
IP Event Dampening

```
router-a(config-if)#dampening 15 1000 1500 60 1000
```



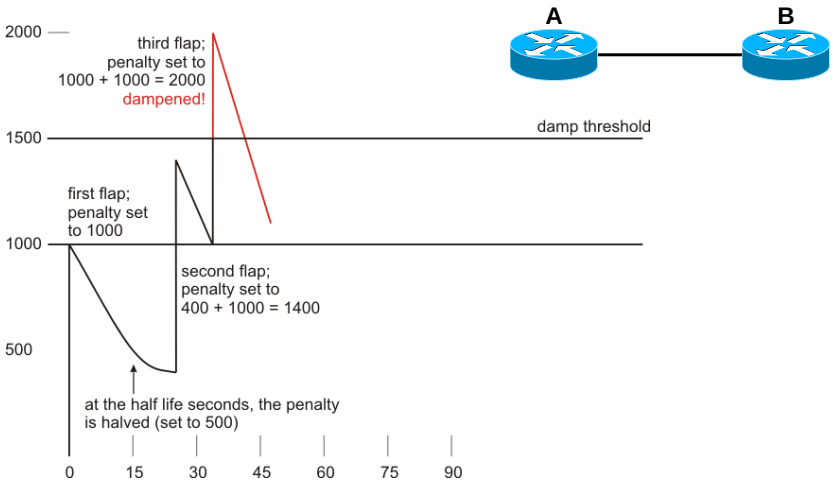
IP Event Dampening

```
router-a(config-if)#dampening 15 1000 1500 60 1000
```



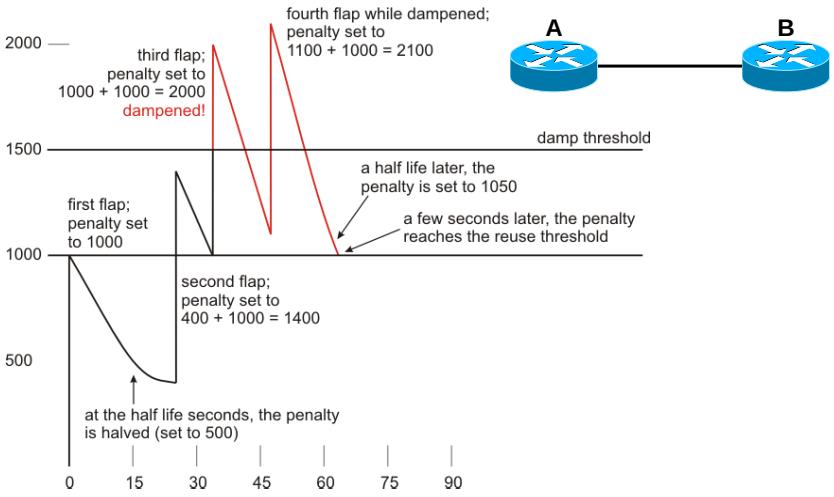
IP Event Dampening

```
router-a(config-if)#dampening 15 1000 1500 60 1000
```



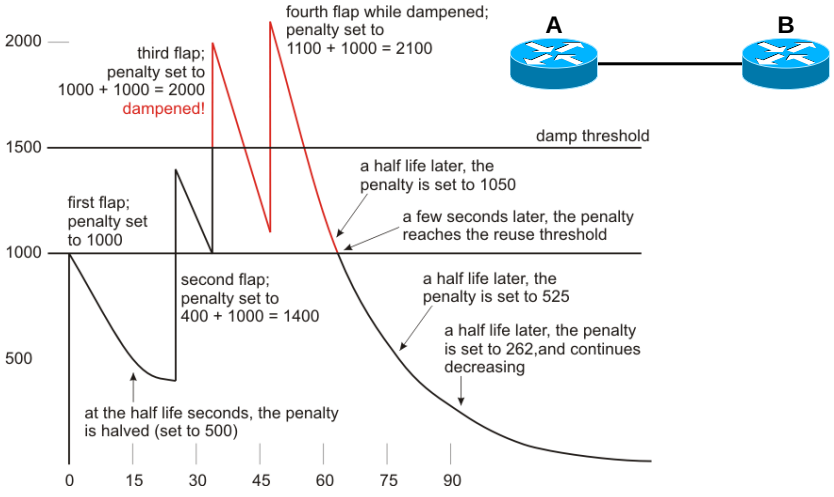
IP Event Dampening

```
router-a(config-if)#dampening 15 1000 1500 60 1000
```



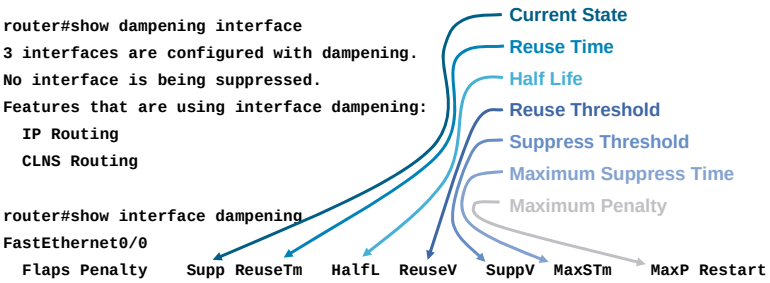
IP Event Dampening

```
router-a(config-if)#dampening 15 1000 1500 60 1000
```



IP Event Dampening

```
router#show dampening interface
3 interfaces are configured with dampening.
No interface is being suppressed.
Features that are using interface dampening:
  IP Routing
  CLNS Routing

router#show interface dampening
FastEthernet0/0
  Flaps Penalty    Supp ReuseTm   HalfL  ReuseV   SuppV  MaxSTm    MaxP Restart
     0       0   FALSE       0       5    1000    2000      20   16000       0
ATM2/0
  Flaps Penalty    Supp ReuseTm   HalfL  ReuseV   SuppV  MaxSTm    MaxP Restart
     0       0   FALSE       0       5    1000    2000      20   16000       0
POS6/0
  Flaps Penalty    Supp ReuseTm   HalfL  ReuseV   SuppV  MaxSTm    MaxP Restart
     0       0   FALSE       0       5    1000    2000      20   16000       0
```

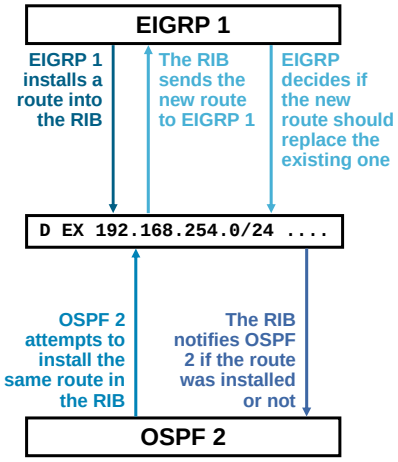


IP Event Dampening

- Prevents routing protocol churn caused by constant interface state changes
- Supports all IP routing protocols
  - Static Routing, RIP, EIGRP, OSPF, IS-IS, BGP
  - In addition, it supports HSRP and CLNS routing
  - Applies on physical interfaces and can't be applied on sub-interfaces individually
- Available in 12.0(22)S Cisco IOS

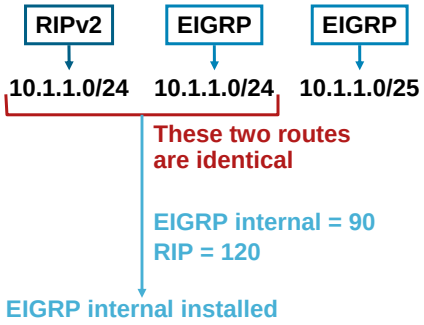
Route Selection

- How does the RIB decide which route is best among various sources?
- It actually doesn't
- Each route is marked with the installing routing process
- When another process attempts to install an overlapping route in the RIB, the RIB allows the owner of the current route to decide if it should be installed or not
- Generally, this decision is made using the administrative distance of the two routing processes



Route Selection

- How is administrative distance used to determine which route should be installed?
- Only identical routes are compared
  - Identical prefixes with different prefix lengths are not the same route
- The route from the protocol with the lower administrative distance is installed



Route Selection

The RIB receives OSPF's new route, calls into EIGRP, and EIGRP determines if the OSPF route should be installed

The RIB receives the EIGRP reply and flushes the EIGRP route

```
RT: closer admin distance for 192.168.239.0, flushing 1 routes
IP-EIGRP(Default-IP-Routing-Table:100): Callback: lostroute
192.168.239.0/24
RT: add 192.168.239.0/24 via 208.0.245.11, ospf metric [110/65]
```

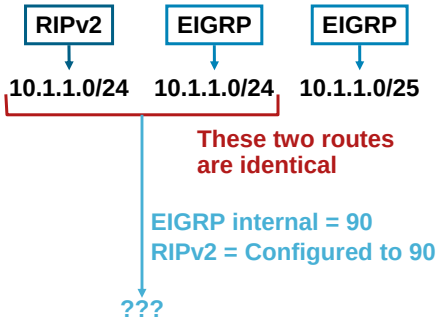
EIGRP receives a callback stating the RIB has removed one of its routes

The RIB installs OSPF's route

The RIB notifies OSPF its route has been installed

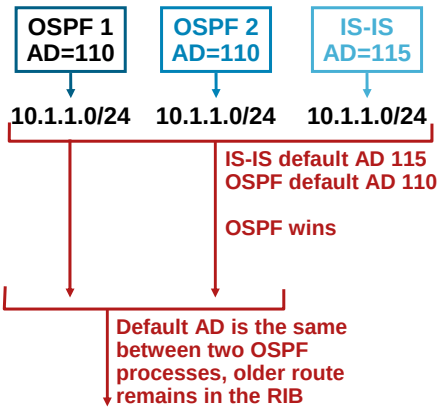
Route Selection

- What happens if the administrative distance of the two routes are equal?
- It depends on the routing protocol



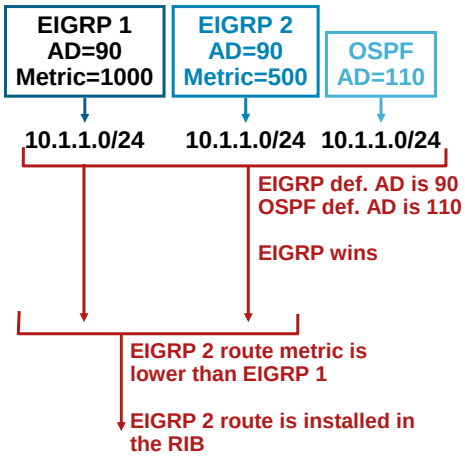
Route Selection

- OSPF and IS-IS
- The default administrative distance of each route is compared
- If these are the same, the older route remains in the routing table



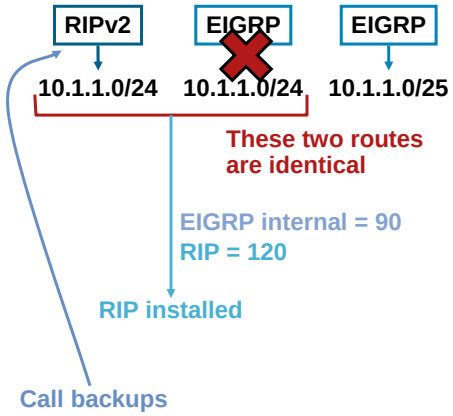
Route Selection

- EIGRP
- Default administrative distance of each route's protocol is compared
- If these are the same, both routes must be EIGRP
- Compare the metric type and metric, the lower cost route is installed
- If the metric and metric type are the same, compare the EIGRP AS number
- The lower AS number wins



Backup Routes

- If a route with a low administrative distance fails…
- The routing table calls each routing process asking for backup routes
- Each routing process attempts to install its matching routes
- The route with the lowest administrative distance wins



Backup Routes

```
router-b#show ip route
Codes: D - EIGRP, EX - EIGRP external, O – OSPF....
....
O    10.0.16.0/24 [110/1064] via 10.0.12.10, Serial0/3
router-b#show ip eigrp topo
IP-EIGRP Topology Table for AS(100)/ID(208.0.17.11)
....
P 10.0.16.0/24, 0 successors, FD is Inaccessible
router-b#debug ip routing
IP routing debugging is on
router-b#debug ip eigrp notifications
IP-EIGRP Event notification debugging is on
....
RT: delete route to 10.0.16.0/24
IP-EIGRP: Callback: callbackup_routes 10.0.16.0/24
IP-EIGRP: Callback: reload_iptable
RT: add 10.0.16.0/24 via 10.0.12.10, eigrp metric [170/3072256]
router-b#show ip route
Codes: D - EIGRP, EX - EIGRP external, O – OSPF....
....
EX   10.0.16.0/24 [170/3072256] via 10.0.12.10, Serial0/3
....
```

The route is installed by OSPF

EIGRP has the same route in its topology table, but it's not installed because it has a higher AD

The OSPF route fails…

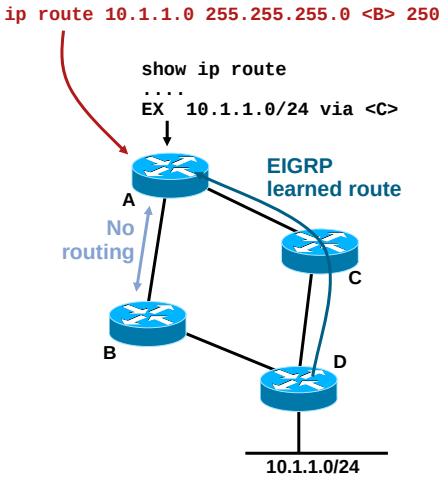
EIGRP gets a callback for 10.0.16.0/24, which is the OSPF route that failed

EIGRP installs the existing 10.0.16.0/24 route from its topology table

The route is now installed by EIGRP

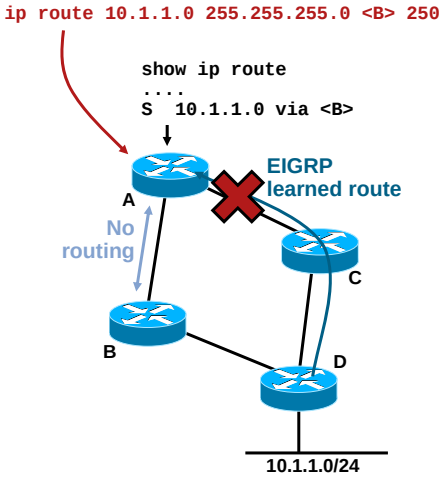
Static Routes

- The concepts of administrative distance and backup routes are used to create floating static routes
- Configuring a static route with a very high administrative distance ensures it won't be installed as long as there's a dynamically learned route installed in the RIB



Static Routes

- When the dynamically learned route fails, the RIB calls the processes, looking for a backup route
- Since no other processes have routes to install, the static route with an administrative distance of 250 wins



Static Routes

Static Routes Can Have a Next Hop of an IP Address

- `ip address 10.1.1.0 255.255.255.0 10.1.2.1`
- This causes the RIB and CEF to recurse to find the correct Layer 2 header to rewrite onto the packet
- For each packet destined to 10.1.1.0/24:

  Look up the destination

  Find the next hop is 10.1.2.1

  Look up 10.1.2.1

  Look up the layer 2 header and interface for the next hop toward 10.1.2.1
- As long as the next hop is reachable, the router assumes the destination through that next hop is reachable

Static Routes

Static Routes Can Have a Next Hop of a Point-to-Point Interface

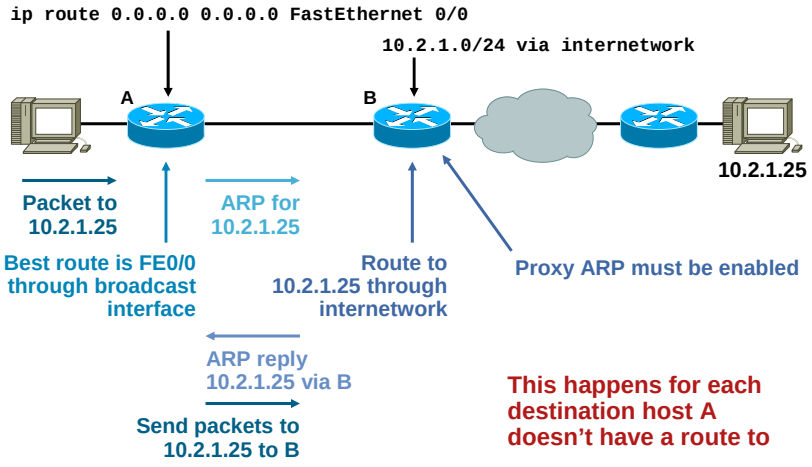
- `ip address 10.1.1.0 255.255.255.0 serial0`
- The RIB and CEF point the route directly at the point-to-point interface
- For each packet destined to 10.1.1.0/24, the Layer 2 rewrite header is set up to reach the other end of the point-to-point link
- This is not complicated, and fast
- As long as the interface is up, the router assumes the destination is reachable through that interface

Static Routes

- Static routes can have a next hop of a broadcast interface
- The RIB and CEF will point this route directly to the broadcast interface

```
router(config)#ip route 10.1.0.0 255.255.0.0 fa 0/0

router#show ip route
....
10.0.0.0/16 is subnetted, 1 subnets
S     10.1.0.0 is directly connected, FastEthernet0/0
```


Static Routes



Static Routes

- For a default route (0.0.0.0/0), this could result in 2^{32} ARP entries in A's local tables
  - This would overflow the ARP cache, and crash A
- Control static routes to broadcast interfaces
  - Small range of reachable addresses
  - Don't use with proxy ARP, just for reaching hosts actually connected to that segment
- Static routes to point-to-point interfaces don't have this problem

Static Routes

- A static route to an interface is shown in the routing table as connected:

```
router(config)#ip route 10.1.0.0 255.255.0.0 fa 0/1

router#show ip route
....
10.0.0.0/16 is subnetted, 1 subnets
S       10.1.0.0 is directly connected, FastEthernet0/1
```

- Static routes to interfaces will be included if you configure redistribute connected
- How do routing protocols handle this in relation to the network statement?

Static Routes

- OSPF:
  - Static routes to interfaces are not advertised as a result of a network statement
- IS-IS:
  - IS-IS doesn't use network statements, so static routes to interfaces are not advertised without redistribution
- EIGRP:
  - Static routes to interfaces are considered connected routes
  - They will be picked up and advertised if they are contained within a network statement
- BGP:
  - Static routes to interfaces are installed the routing table
  - They will be picked up and advertised if they match a network statement

Discard Routes

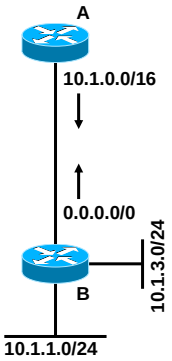
- Discard routes are created when a router aggregates routing information

```
(EIGRP) ip summary-address eigrp 100 10.1.0.0 255.255.0.0 5
(OSPF)  area 1 range 10.1.0.0 255.255.0.0
(IS-IS) summary-address 10.1.0.0 255.255.0.0 level-2
....
2651A#show ip route
....
D       10.1.0.0/16 is a summary, 00:04:03, Null0
```

- A discard route has an administrative distance of five by default

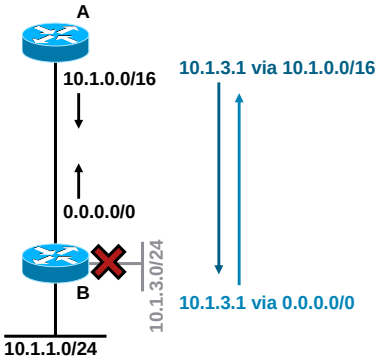
Discard Routes

- Why is this discard route created?
- Suppose
  - A is advertising a default route toward B
  - B is installing this route, so it reaches all unknown networks through A
  - B is advertising the summary 10.1.0.0/16 to A
  - B is not building a discard route for this summary
  - 10.1.3.0/24 is no longer reachable through B (the network has been disconnected)



Discard Routes

- A receives a packet for 10.1.3.1
  - A examines its local routing table, and finds the best path is through B, using the route to 10.1.0.0/16
  - A forwards the packet to B
- B receives the packet for 10.1.3.1
  - B examines its local routing table, and finds the best path is through A, using the default route
  - B forwards the packet to A
- We have a permanent routing loop!
- If B builds a discard route for 10.1.0.0/16, it will discard the packet, rather than forwarding it through the default route back to A



Discard Routes

- Can you prevent the routing protocol from creating a discard route?

  OSPF

```
router ospf 100
 no discard route
```

  EIGRP

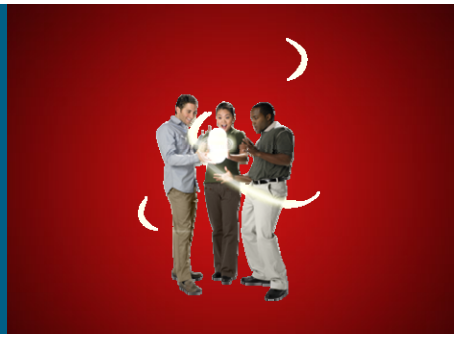
```
interface serial0
 ip summary-address 10.1.0.0 255.255.0.0 255
```

- Why would you want to get rid of the discard route?

  A summary without a discard route is effectively a filter for all the summary components

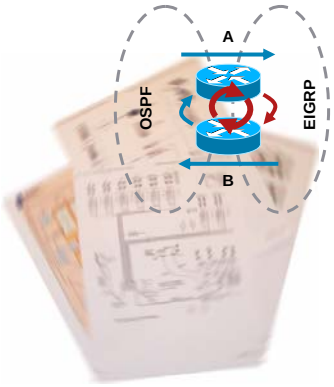
  Be very careful with removing discard routes—this can create routing loops

Route Distribution



Route Redistribution

- Redistribution Fundamentals
- Redistribution Filters

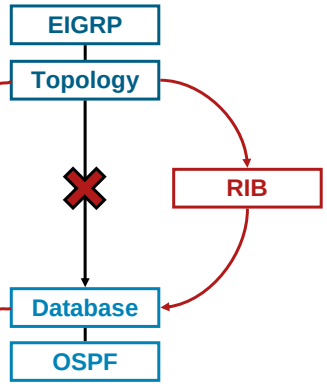


Redistribution Fundamentals

- Routes can be redistributed from one routing protocol to another
- Routes really aren't redistributed between protocols

  Routes are taken from the RIB, not another protocol!

  The redistributing protocol knows which routes to take from the RIB based on the "known via" information




```
router#show ip route 10.0.0.0
....
Routing entry for 10.0.0.0/8
  Known via "eigrp 100", distance 90, metric 3072256, type internal
  Redistributing via eigrp 100
```

Redistribution Fundamentals

- A route must be installed in the RIB for it to be redistributed
- Routes redistributed from the routing table are not installed by the pulling protocol

  EIGRP and RIP will advertise a route not installed in the RIB by the redistributing protocol in this case

  This is contrary to the normal distance vector protocol rules about advertising routes not installed by the advertising process
- Anything not in the RIB can't be used for filtering redistribution

Redistribution Filters

- Filters can control what information is injected into a routing protocol through redistribution
- Filters can also be used to stop routing loops when mutual redistribution between two routing protocols is configured
- What information about installed routes does the RIB have (what information can we filter on)?

Redistribution Filters

Route Maps Can Filter On These Fields in the RIB:

- `match metric`
  - Filtering is possible between all protocols based on metric
  - It's very tricky, as the metric must be exact to match
- `match tag`
  - Some protocols carry route tags: EIGRP, OSPF externals, IS-IS
  - BGP can match tags local to the system but does not carry them
  - This normally works well, but is sometimes tricky, if the protocol can carry more than one tag
- `match ip address`
  - Matches the prefix, possibly the network mask, depending on the access list type used, etc.
  - More on this in the section on route filtering, later in this presentation

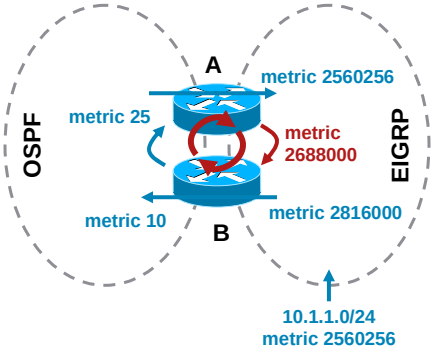
Redistribution Filters

Route Maps Can Filter On These Fields in the RIB:

- `match ip next-hop`
  - Matches on the next hop listed in the routing table
- `match route-type`
  - Internal: EIGRP or OSPF internal routes
  - External: EIGRP or OSPF external routes, type-1 and type-2 options for OSPF
  - Level-1 or level-2: IS-IS route levels
- `match interface`
  - The interface through which the router is forwarding traffic to the destination, as listed in the RIB

Redistribution Filters

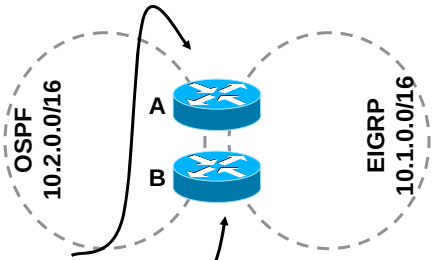
- A route is injected into EIGRP as an external; this route is redistributed through B into OSPF
- The route is transmitted to A through OSPF, and redistributed into EIGRP
- The metric is set manually in redistribution at A to something lower than the original external injected into EIGRP
- B prefers this route, building a routing loop



Redistribution Filters

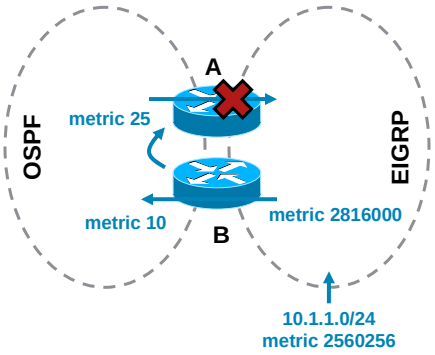
- To filter based on prefixes, configure route-maps, which match the address ranges using access-lists used by each section of the network

```
access-list 10 permit 10.1.0.0 0.0.255.255
access-list 20 permit 10.2.0.0 0.0.255.255

route-map test1 permit
  match ip address 10
route-map test2 permit
  match ip address 20

router eigrp 100
 redistribute ospf 100 metric 1000 1 255 1 1500 route-map test2

router ospf 100
 redistribute eigrp 100 metric 10 route-map test1
```



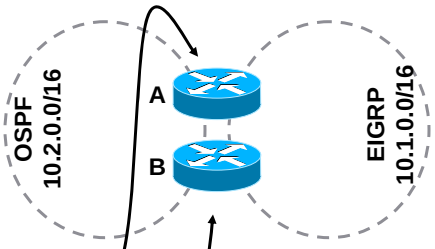
Redistribution Filters

- A route is injected into EIGRP as an external; this route is redistributed through B into OSPF
- The route is transmitted to A through OSPF, and redistributed into EIGRP
- The route is now blocked by route-map test2, which breaks the routing loop



Redistribution Filters

- EIGRP and OSPF can set tags on their external routes
- Set the tag when redistributing between the protocols. Deny tagged routes at the redistribution point

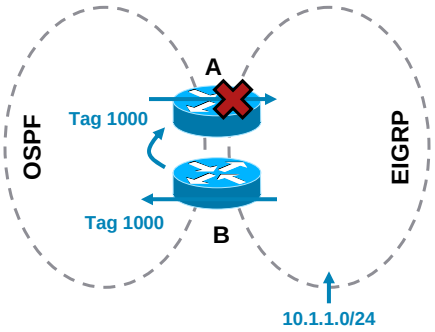



```
route-map cisco deny 10
 match tag 1000
route-map cisco permit 20
 set tag 1000
....
router eigrp 100
 redistribute ospf 100 metric 1000 1 255 1 1500 route-map cisco
 ....
router ospf 100
 redistribute eigrp 100 metric 10 route-map cisco
```

Redistribution Filters

- 10.1.1.0/24 route is injected into EIGRP as an external; it is redistributed through B into OSPF, and tagged
- 10.1.1.0/24 is transmitted to A through OSPF; the route is blocked from being redistributed into EIGRP because of the route tag
- IS-IS can also tag and filter based on tags

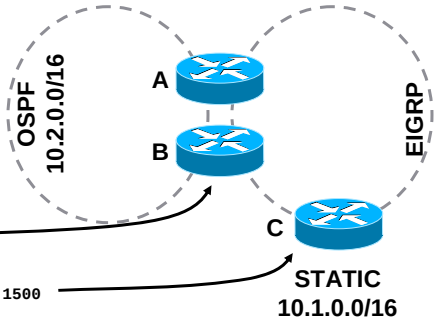
  draft-ietf-isis-admin-tags



Redistribution Filters

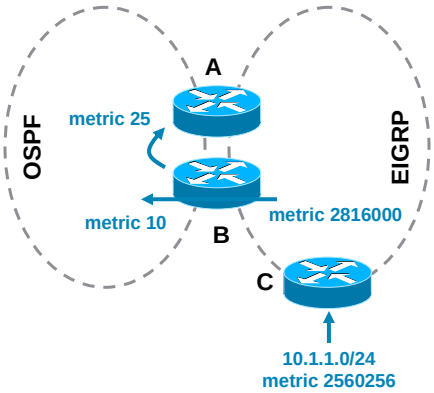
- If live routing data is only needed in one direction (normally, this is true), redistribute a static in one direction, and between protocols in the other direction

```
ip route 10.1.0.0 255.255.0.0 serial 0/0
....
router ospf 100
 redistribute eigrp 100 metric 10
 ....
router eigrp 100
 redistribute static 100 metric 1000 1 255 1 1500
 ....
```

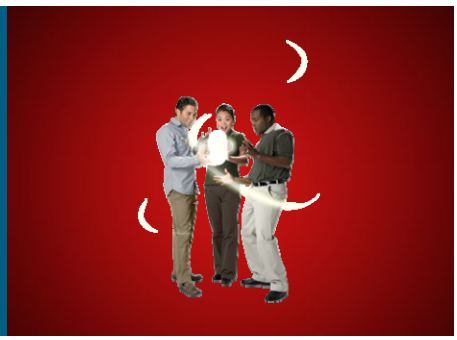


Redistribution Filters

- A route is injected into EIGRP as an external through C; this route is redistributed through B into OSPF
- The route is transmitted to A through OSPF; the route is not redistributed back into EIGRP, since redistribution between OSPF and EIGRP is not configured



Filtering Routes

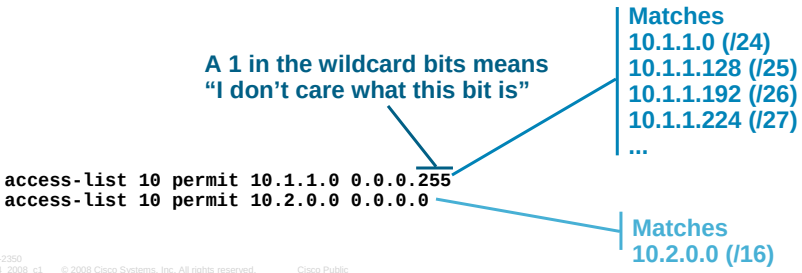


Filtering Routes

- Standard Access Lists
- Prefix Lists
- Route Maps

Standard Access Lists

- Standard access lists are numbered between 1 and 99
- The single expression matches against the prefix portion of the route
- The network mask (prefix length) is ignored when filtering using standard access lists



Standard Access Lists

These bits are 0's, so they must be identical in the match and input networks	These bits are 1's, so the input network doesn't need to match the match network	
`00000000 00000000 00000000`	`11111111: 0.0.0.255`	**Wildcard Bits**
`00001010 00000001 00000001`	`00000000: 10.1.1.0`	**Match**
These match	**These don't matter**	
`00001010 00000001 00000001`	`11000000: 10.1.1.192`	**Input Network 1**
These don't match	**These don't matter**	
`00001010 00000001 00000010`	`00000000: 10.1.2.0`	**Input Network 2**

Standard Access Lists

Network	Wildcard	Matches
`10.1.1.0`	`0.0.0.255`	Any network in the 10.1.1.0/24-10.1.1.255/32 space
`10.1.1.0`	`0.0.0.127`	10.1.1.0/25-32, 10.1.1.128/25-32
`10.1.1.0`	`0.0.0.63`	10.1.1.0/26-32, 10.1.1.64/26-32, 10.1.1.128/26-32, 10.1.1.192/26-32
`10.1.1.0`	`0.0.0.31`	10.1.1.0/27-32, 10.1.1.32/27-32, 10.1.1.64/27-32, 10.1.1.96/27-32, 10.1.1.128/27-32, 10.1.1.160/27-32, 10.1.1.192/27-32, 10.1.1.224/27-32
`10.1.1.4`	`0.0.0.1`	10.1.1.4/31-32, 10.1.1.5/32
`10.1.1.64`	`0.0.0.3`	10.1.1.64/30-32, 10.1.1.68/30-32
`10.1.1.96`	`0.0.0.7`	10.1.1.96/29-32, 10.1.1.104/29-32, 10.1.1.112/29-32, 10.1.1.120/29-32

Prefix Lists

- Prefix lists are easier to filter routes with (it might look harder, but it's not)
- ip prefix-list list-name [seq seq-value] {permit|deny} network/length [ge value] [le value]
  - Named, rather than numbered
  - Optional sequence allows you to modify, remove, or add individual lines in the prefix list
  - Network is the prefix
  - Length is the number of bits in the network mask
- ge is greater than or equal to (>=)
- le is less than or equal to (<=)

Prefix Lists

`permit 10.1.1.0/24`	10.1.1.0/24 only, does not match 10.1.1.0/25-32
`permit 10.1.1.0/24 ge 25`	Any /25 or longer within 10.1.1.0/24 e.g. 10.1.1.0/25-32, 10.1.1.2/31, 10.1.1.4/30-31, 10.1.1.254/31, etc.
`permit 10.1.1.0/24 le 30`	Any /30 or shorter within 10.1.1.0/24 e.g. 10.1.1.0/24-30, 10.1.1.4/30, 10.1.1.128/25-30, etc.
`permit 10.1.1.0/24 ge 25 le 30`	Any /25 or longer and /30 or shorter within 10.1.1.0/24 e.g. 10.1.1.0/25-30, 10.1.1.4/30, 10.1.1.128/25-30, etc.
`permit 0.0.0.0/0 ge 8 le 24`	Any /24 or shorter and /8 or longer in any address space

Prefix Lists

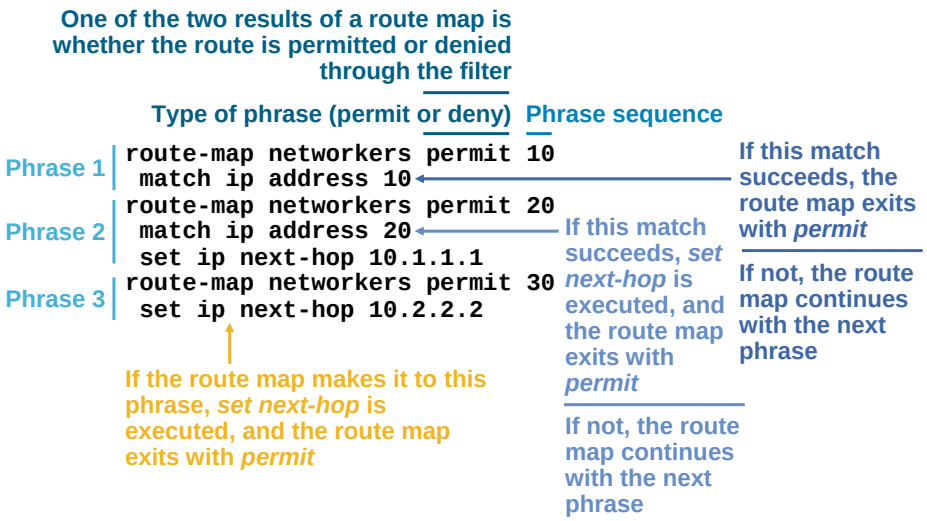
Where Can You Use Prefix Lists?

- BGP routes advertised to a specific neighbor or from a specific neighbor, through the neighbor statement
- OSPF type 3 route filters at an ABR
- EIGRP routes using distribute-list
- Route maps

Route Maps

- Route maps allow you to:
  - Combine more than one type of filter into a single phrase
  - Use some rudimentary forms of logical “AND” and “OR” to filter routes
  - Set some route attributes, rather than just permitting or denying routes
- Route maps can be used to:
  - Set IP next-hop (Policy routing)
  - Filter BGP
  - Filter EIGRP
  - Filter routes being redistributed between two protocols
  - Etc.

Route Maps



Route Maps

Phrase type	Match result	Route map result
Permit	Permit	Set statements within the phrase are executed, route map exits with *permit*
Permit	Deny	Set statements within phrase are not executed, route map continues with next phrase *If there is no next phrase, route map exits with deny*
Deny	Permit	Set statements within phrase are not executed, route map exits with *deny*
Deny	Deny	Set statements within phrase are not executed, route map continues with the next phrase *If there is no next phrase, route map exits with deny*

Route Maps

Match	Description
`metric`	Metric of the route *In BGP's case, this is the MED* *Must match exactly!*
`route-type`	OSPF or EIGRP route type *Internal, External OSPF external type 1 or 2*
`tag`	Route tag
`ip address`	Standard or extended access list *Applied against the prefix* *Numbered or named*
`ip address prefix-list`	Prefix list *Applied against the prefix and prefix length*
`ip next-hop`	Standard or extended access list *Applied against the next hop (via in the routing table)* *Numbered or named*
`ip route-source`	Standard or extended access list *Applied against the neighbor this route was learned from (from in the routing table)* *Numbered or named*

Route Maps

- Not all set statements work with all protocols or in all situations
- There is no definitive list of what works where
- The best thing to do is to test what you want to do before you try and use it

Set	Description
`ip next-hop`	Set the next hop in the routing table or transmitted route
`metric`	Set the metric of the redistributed or transmitted route
`metric-type`	Set the type of external route *External type 1 or type 2 for OSPF*
`tag`	Sets the route tag

Route Maps

Policy-Based Routing (PBR)

- Allows packets to be filtered through route maps containing policies that selectively determine the next hop to which packets are to be forwarded
- Policy routes can be determined based on such things as the source of the packet, protocol types, port numbers, and the size of the packet
- Must be applied on the interface on which the packet is received. “ip policy route-map <name>” in interface configuration mode

Route Maps

Benefits of Policy-Based Routing

- Load sharing—Supplemental to dynamic load-sharing capabilities offered by Cisco IOS, PBR allows traffic to be administratively distributed among multiple paths based on the traffic characteristics
- Quality of Service (QoS)—Using IP Precedence or type of service (ToS) values to prioritize differentiated traffic
- Source-sensitive routing—Route traffic originating from different users through different paths
- Cost—Route traffic across low-bandwidth, low-cost permanent paths or high-bandwidth, high-cost, switched paths

Route Maps

Policy-Based Routing Match Commands

Command	Description
`match ip address access-list number | name`	Matches a packet with the characteristics specified in the standard or extended access lists
`match length min max`	Matches the Layer 3 length

Route Maps

Policy-Based Routing Set Commands

Command	Description
`set default interface type number`	Set the outgoing interface for matched packets when there is no explicit route to the destination
`set interface type number`	Sets the outgoing interface for matched packets when there is an explicit route to the destination
`set ip default next-hop ip-address`	Sets the next-hop router address for matched packets when there is no explicit router to the destination
`set ip next-hop ip-address`	Sets the next-hop router address for the matched packets when there is an explicit route to the destination
`set ip precedence precedence`	Sets the precedence bits in the Type of Service field of matched packets
`set ip tos type-of-service`	Sets the ToS bits in the Type of Service field of matched packets

Route Maps

Policy Routing vs. Destination-Based Routing

- PBR proceeds through the route map until a match is found. If no match is found in the route map, the packet will be forwarded according to normal destination-based routing
- If the route-map statement is marked as a deny, the packets meeting the match criteria are forwarded according to normal destination-based routing
- If the statement is marked as permit and the packets do not meet the match criteria, the packets are forwarded according to normal destination-based routing
- If the route-map statement is marked as permit and the packets meet the match criteria, the set clauses are applied and policy routing is performed

Route Maps

Route map	Logic	Notes

`route-map networkers permit 10`
`match ip address 10`
`match tag 1000`
`set ip next-hop 10.1.1.1`
	AND	Both matches must succeed for the set to be executed
		Some types of matches cannot co-exist in the same route map phrase, such as an access list and a prefix list

`route-map networkers permit 10`
`match ip address 10 20`
`set ip next-hop 10.1.1.1`
	OR	If the route matches either access list 10 or 20, the set will be executed

`route-map networkers permit 10`
`match ip address 10`
`route-map networkers permit 20`
`match ip address 20`
`route-map networkers permit 30`
`set ip next-hop 10.1.1.1`
	AND	The route must not match access list 10 or 20 for the set to execute
		If the access lists deny routes, then the routes must not exist for a specific action to be taken (useful in conditional advertisement)

Route Maps

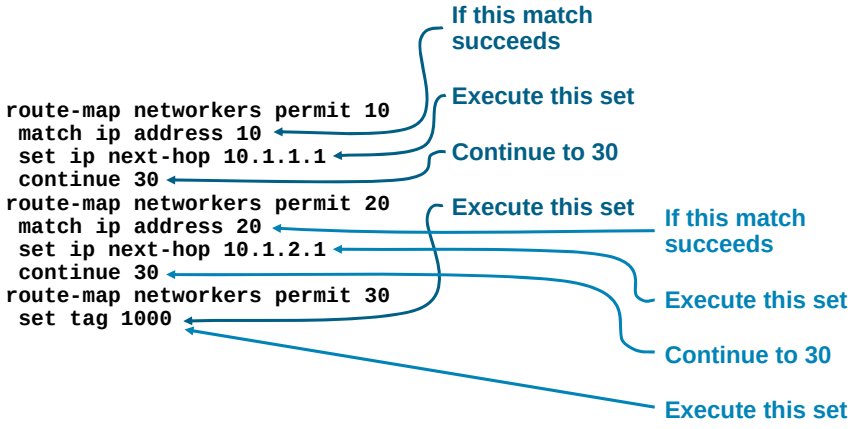
- Route map AND and OR rules
  - If two different types of matches are configured in the same phrase, they must both succeed for the set to be executed and the route map to exit (logical AND)
  - If two of the same type of match are configured (where permitted) in the same phrase, the set will be executed and the route map will exit (logical OR)
  - All matches in a single phrase must fail for the route map to fall through to the next phrase (logical AND)
- Route maps can become very complicated based on these parsing rules

Route Maps

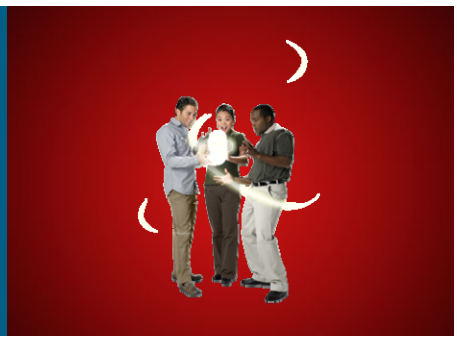
- In normal processing, if all matches fail, the route map falls through to the next phrase
- Route map continue allows you to continue to another phrase if the matches succeed
- Sets are executed before the continue is followed
- Use for:
  - Gathering policy (matches and sets) into a single phrase
  - More complex logical constructions

Route Maps

Gathering Policy with Continue



Load Sharing



Load Sharing

- Assume the same routing process attempts to install two routes for the same destination in the RIB
- The routing process may allow the second route to be installed based on its own rules

	OSPF	IS-IS	EIGRP
Route cost	Must be equal to installed route	Must be equal to installed route	Must be less than the variance times the lowest cost installed route
Maximum Paths	Must be fewer than *maximum-paths* configured under the routing process		

Load Sharing

```
router#show ip route 192.168.239.0
Routing entry for 192.168.239.0/24
  Known via "eigrp 100", distance 170, metric 3072256, type external
  Redistributing via eigrp 100
  Last update from 192.168.245.11 on Serial3/1, 00:18:17 ago
  Routing Descriptor Blocks:
  * 192.168.246.10, from 192.168.246.10, 00:18:17 ago, via Serial3/0
      Route metric is 3072256, traffic share count is 1
      ....
    192.168.245.11, from 192.168.245.11, 00:18:17 ago, via Serial3/1
      Route metric is 3072256, traffic share count is 1
      ....
```

The *traffic share count* is critical to understanding the actual load sharing of packets using these two routes

How is this calculated?

Load Sharing

```
router#show ip route 192.168.239.0
Routing entry for 192.168.239.0/24
  Known via "eigrp 100", distance 170, metric 3072256, type external
  Redistributing via eigrp 100
  Last update from 192.168.245.11 on Serial3/1, 00:18:17 ago
  Routing Descriptor Blocks:
  * 192.168.246.10, from 192.168.246.10, 00:18:17 ago, via Serial3/0
      Route metric is 3072256, traffic share count is 1
      ....
    192.168.245.11, from 192.168.245.11, 00:18:17 ago, via Serial3/1
      Route metric is 3072256, traffic share count is 1
      ....
```

The metric of each route is divided into the highest metric among the available metrics

3072256/3072256 == 1

The resulting number is the traffic share count

Load Sharing

```
router#show ip route 192.168.239.0
Routing entry for 192.168.239.0/24
  Known via "eigrp 100", distance 170, metric 3072256, type external
  Redistributing via eigrp 100
  Last update from 192.168.245.11 on Serial3/1, 00:18:17 ago
  Routing Descriptor Blocks:
  * 192.168.246.10, from 192.168.246.10, 00:18:17 ago, via Serial3/0
      Route metric is 1536128, traffic share count is 2
      ....
    192.168.245.11, from 192.168.245.11, 00:18:17 ago, via Serial3/1
      Route metric is 3072256, traffic share count is 1
      ....
```

If the lower metric is less than the second metric, the traffic share count will be something other than 1 (only for EIGRP and requires variance to be configured)

3072256/3072256 == 1

3072256/1536128 == 2

The resulting number is the traffic share count

Load Sharing

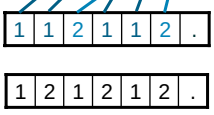
```
router#show ip route 192.168.239.0
Routing entry for 192.168.239.0/24
  Known via "eigrp 100", distance 170, metric 3072256, type external
  Redistributing via eigrp 100
  Last update from 192.168.245.11 on Serial3/1, 00:18:17 ago
  Routing Descriptor Blocks:
  * 192.168.246.10, from 192.168.246.10, 00:18:17 ago, via Serial3/0
      Route metric is 1536128, traffic share count is 2
      ....
    192.168.245.11, from 192.168.245.11, 00:18:17 ago, via Serial3/1
      Route metric is 3072256, traffic share count is 1
      ....
```

When process switching, *traffic share count* packets is sent down one path, and then the process moves to the next available path

The route with the * beside it is the current in use path

Load Sharing

```
router#show ip route 192.168.239.0
Routing entry for 192.168.239.0/24
  Known via "eigrp 100", distance 170, metric 3072256, type external
  Routing Descriptor Blocks:
  * 192.168.246.10, from 192.168.246.10, 00:18:17 ago, via Serial3/0
      Route metric is 1536128, traffic share count is 2
      ....
    192.168.245.11, from 192.168.245.11, 00:18:17 ago, via Serial3/1
      Route metric is 3072256, traffic share count is 1
      ....
```



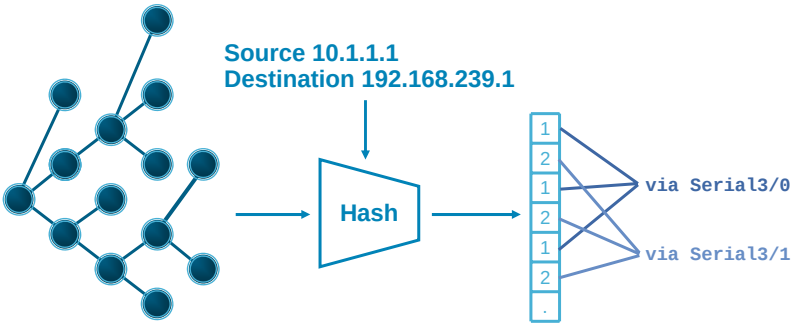
Equal cost paths

CEF uses the traffic share count to fill in the load share table

Each path is placed in the table the number of times its traffic share count indicates, and then the next path is inserted, etc, until the load share table is full

The size of the load share table is platform dependant

Load Sharing



CEF finds the prefix in the CEF trie

CEF hashes the source and destination addresses, and chooses a bucket from the load share table

The load share table points to an adjacency corresponding to one of the next hops in the routing table

Load Sharing

```
router#show ip cef 192.168.239.0
192.168.239.0/24, version 127, epoch 0, per-destination sharing
0 packets, 0 bytes
  via 192.168.246.10, Serial3/0, 0 dependencies
    traffic share 1
    next hop 192.168.246.10, Serial3/0
    valid adjacency
  via 192.168.245.11, Serial3/1, 0 dependencies
    traffic share 1
    next hop 192.168.245.11, Serial3/1
    valid adjacency
  0 packets, 0 bytes switched through the prefix
  tmstats: external 0 packets, 0 bytes
           internal 0 packets, 0 bytes
```

Two equal cost paths

Traffic share counters

```
router#show ip cef exact-route 10.1.1.1 192.168.239.1
10.1.1.1        -> 192.168.239.1    : Serial3/0 (next hop 192.168.246.10)
```

Load Sharing

```
router#show ip cef 192.168.239.0 internal
192.168.239.0/24, version 9, epoch 0, per-destination sharing
0 packets, 0 bytes
  via 192.168.246.10, Serial3/0, 0 dependencies
    traffic share 1
    next hop 192.168.246.10, Serial3/0
    valid adjacency
  via 192.168.245.11, Serial3/1, 0 dependencies
    traffic share 1
    next hop 192.168.245.11, Serial3/1
    valid adjacency
0 packets, 0 bytes switched through the prefix
tmstats: external 0 packets, 0 bytes
internal 0 packets, 0 bytes
Load distribution: 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 (refcount 1)
Hash  OK  Interface                 Address         Packets
1    Y   Serial3/0                 point2point           0
2    Y   Serial3/1                 point2point           0
3    Y   Serial3/0                 point2point           0
4    Y   Serial3/1                 point2point           0
.
.
```


Load Sharing

CEF Has Two Forms of Load Sharing

- Per-session load sharing (per-destination)
- Per-packet load sharing

Load Sharing

Per-Session Load Sharing

- Often referred to as per-destination load sharing, even within Cisco IOS
- This method is the default behavior and does not require any additional configuration
- A session is a flow that shares the same source and destination. Traffic with different source to destination pairs tend to take different paths
- This method ensures that traffic for a given session arrive in order
- Has the potential for traffic polarization and is more effective as the number of source to destination pairs increase

Load Sharing

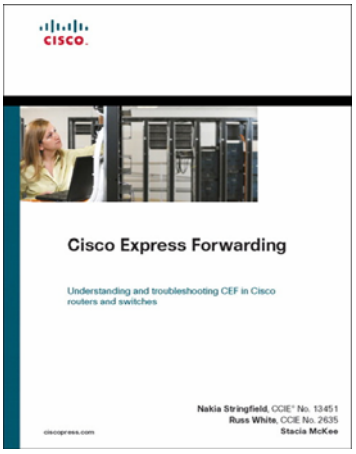
Per-Packet Load Sharing

- To utilize this method, configure “ip cef load-sharing per-packet” in interface configuration mode. Each outgoing interface must have this command configured
- Uses a round-robin method to determine which path each packet takes to the destination without consideration of source to destination sessions
- Ensures traffic is more evenly distributed over multiple paths
- Packets for a given source to destination session may take different paths, introducing a greater potential for packets to arrive out of sequence. Not advisable for all types of traffic

Q and A

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.