Deploying Cisco IOS® IP Routing Features

Session RST-206

Agenda

- Background
- Host Interaction
- Too Much Information
- Multiple Routing Protocols
- Redistribution
- Policy Routing
Background

CCIE Credo

“Just because you can, doesn’t mean you should.”
### Router Functions

- **Routing** = building maps and giving directions
- **Switching** = forwarding packets between interfaces
- Routers are packet relays or “switches”
- Path determination is overhead

### Routing Protocols

- Routers are packet switches that forward traffic based on layer 3 logical addresses
- Routing protocol updates are exchanged by routers to learn about paths to other logical networks
- Each routing protocol offers features that can make it desirable as part of an internetwork design
### Internet Routing Protocols

- IP routing protocols are characterized as

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Function</th>
<th>Updates</th>
<th>Converge</th>
<th>Metric</th>
<th>VLSM</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>RIP</td>
<td>DV</td>
<td>IGP</td>
<td>30 Sec</td>
<td>90 Sec</td>
<td>Hops</td>
<td>No</td>
<td>Auto</td>
</tr>
<tr>
<td>RIPv2</td>
<td>DV</td>
<td>IGP</td>
<td>30 Sec</td>
<td>90 Sec</td>
<td>Hops</td>
<td>Yes</td>
<td>Auto</td>
</tr>
<tr>
<td>IGRP</td>
<td>DV</td>
<td>IGP</td>
<td>30 Sec</td>
<td>270 Sec</td>
<td>Compound</td>
<td>No</td>
<td>Auto</td>
</tr>
<tr>
<td>EIGRP</td>
<td>Adv DV</td>
<td>IGP</td>
<td>Triggered</td>
<td>&lt; 5 Sec</td>
<td>Compound</td>
<td>Yes</td>
<td>Both</td>
</tr>
<tr>
<td>OSPF</td>
<td>LS</td>
<td>IGP</td>
<td>Triggered</td>
<td>&lt; 5 Sec</td>
<td>Cost</td>
<td>Yes</td>
<td>Manual</td>
</tr>
<tr>
<td>ISIS</td>
<td>LS</td>
<td>IGP</td>
<td>Triggered</td>
<td>&lt; 5 Sec</td>
<td>Cost</td>
<td>Yes</td>
<td>Both</td>
</tr>
<tr>
<td>BGP</td>
<td>Path Vec</td>
<td>EGP</td>
<td>Incremental</td>
<td>&lt; 60 sec</td>
<td>Path</td>
<td>Yes</td>
<td>Manual</td>
</tr>
</tbody>
</table>

### Host Interaction
How Hosts Transmit

My Address: 10.1.1.2
My Mask: 255.255.255.0
My Network: 10.1.1.2 <AND> 255.255.255.0 == 10.1.1.0

Destination Address: 10.1.1.3
My Mask: 255.255.255.0
Destination Network: 10.1.1.3 <AND> 255.255.255.0 == 10.1.1.0

Destination is on my network;
ARP for layer 2 address of host

Destination Address: 10.1.2.3
My Mask: 255.255.255.0
Destination Network: 10.1.2.3 <AND> 255.255.255.0 == 10.1.2.0

My Network != Destination Network
Destination is on not my network;
send to default gateway (router);
ARP for layer 2 address of router

Proxy ARP

ARP for 10.1.2.2
Respond to ARP
Packet for 10.1.2.2

• Router responds to ARPs for off subnet addresses if it has a route
• Enabled by default
• RFC 1027
ICMP Redirects

- Cisco routers send ICMP redirects when:
  - The input interface is the output interface and...
  - The (sub)network of the source IP address is the same (sub)network of the next-hop IP address of the routed packet and...
  - The datagram is not source-routed and...
  - The system is configured to send redirects; (on by default) you can use the interface subcommand `no ip redirects` to disable ICMP redirects.
  - Unless HSRP is configured on the interface

Find a Default Router

- From a file or DHCP or dynamically:
  - IRDP—ICMP Router Discovery Protocol, RFC 1256
  - Routers periodically announce via ICMP that they are default
  - Clients can solicit routers as well
IRDP on Routers

- Announcements have a lifetime and preference
- Configured per interface; off by default
- Can advertise via all systems multicast (224.0.0.1)
- Preference level can be set

```
ip irdp [multicast
  holdtime seconds (3X max)
  maxadvertinterval seconds (600)
  minadvertinterval seconds (3/4 X max)
  preference number (0)
  address address [number]]
```

IRDP on Hosts

- in.rdisc in Solaris (multicast only)
- Gated in Linux, HP-UX and AIX
  routerdiscovery client yes | no | on | off;
- WinSock2 in Windows
  NT 4.0 KB Article Q223756
  HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\adaptername\Parameters\Tcpip\PerformRouterDiscovery
- DHCP option 31
HSRP—Hot Standby Router Protocol

- Transparent failover of default router
- “Virtual” router created
- One router is active, responds to virtual L2 and L3 addresses
- Others monitor and take over virtual addresses

HSRP-RFC 2281

- HSR multicasts hellos every 3 sec with a default priority of 100
- HSR will assume control if it has the highest priority and preempt configured after delay (default=0) seconds
- HSR will deduct 10 from its priority if the tracked interface goes down
HSRP

- Hot Standby Router Protocol

Router 1:

```
interface ethernet 0/0
ip ospf cost 10
ip address 169.223.10.1 255.255.255.0
standby 10 ip 169.223.10.254
```

Router 2:

```
interface ethernet 0/0
ip ospf cost 6
ip address 169.223.10.2 255.255.255.0
standby 10 priority 150 preempt delay min 10
standby 10 ip 169.223.10.254
standby 10 track serial 0 60
```

HSRP Load Balancing

Router 1:

```
interface ethernet 0/0
standby ip 10 10.1.1.100
standby 10 priority 10 preempt
standby ip 20 10.1.1.200
standby 20 priority 20 preempt
```

Router 2:

```
interface ethernet 0/0
standby ip 10 10.1.1.100
standby 10 priority 20 preempt
standby ip 20 10.1.1.200
standby 20 priority 10 preempt
```

Even Hosts default gateway is 10.1.1.100
Odd Hosts default gateway is 10.1.1.200
HSRP with ICMP Redirects

**Problem:**
Client sends to default R1; R1 send redirect toward R4; Client caches real addresses

**Solution:**
R1 snoops on all HSRP packets; R1 sources from virtual 1 and redirects toward virtual 3

**Notes:**
Redirect to R7 is allowed
Redirects to R8 not allowed because it is passive
standby redirects {enable | disable}

Host Routing

• Configure a static default gateway on the host and HSRP on the routers
• Alternatively, run a discovery protocol (IRDP)
• Run routing process on hosts as last resort
**Speak RIP to Hosts**

- Using EIGRP but hosts run RIP
- Router can:
  - RIP out, but not in
  - Advertise default only

```
router rip
network 172.16.0.0
redistribute eigrp 1 subnets
router eigrp 1
network 172.16.0.0
```

**IP Directed Broadcasts**

**Default Behavior**

```
10.1.2.255 forward drop X
```

```
10.1.2.255 forward 255.255.255.255
```

with ip directed broadcast Configured
**IP Helper Address**

**Default Behavior**

255.255.255.255 → drop X

10.1.1.0/24 → 10.1.2.0/24

255.255.255.255 → 10.1.2.1 forward

with ip helper address enabled

```
interface ethernet 0
ip helper-address 10.1.2.1
```

---

**IP Forward Protocol**

- Flooded UDP packets have destination address changed to ip broadcast-address
- ip forward-protocol spanning-tree
  Uses spanning tree database for flooding
- ip forward-protocol turbo-flood
  Speed-up if using spanning tree flooding

**Example:**

```
ip forward-protocol spanning-tree
bridge 1 protocol dec
access-list 201 deny 0x0000 0xFFFF
interface ethernet 0
bridge-group 1
bridge-group 1 input-type-list 201
```
**UDP Broadcast Application**

- Feed network provides data
- TIC servers UDP broadcast data
- Feed network connected to routers for management

**Helper Addresses**

- IP helper added to router interfaces on TIC network
- Each router sees the other routers broadcasts
- Each station receives multiple copies of data
### UDP Forward Protocol

- Configure spanning tree
- Filter non-routed protocols
- STP path costs set
  - A = 100 default
  - B = 50
- Router A default router
  - IRDP preference

### Router A Configuration

```
ip forward-protocol spanning-tree
ip forward-protocol udp 111
!
interface ethernet 0
  ip address 200.200.200.61 255.255.255.0
  ip broadcast-address 200.200.200.255
!
interface ethernet 1
  ip address 164.53.7.61 255.255.255.192
  ip broadcast-address 164.53.7.63
  ip irdp preference 100
  bridge-group 1
    bridge-group 1 input-type-list 201
  access-list 201 deny 0xFFFF 0x0000
```
Router B Configuration

ip forward-protocol spanning-tree
ip forward-protocol udp 111
!
interface ethernet 0
  ip address 200.200.200.62 255.255.255.0
  ip broadcast-address 200.200.200.255
!
interface ethernet 1
  ip address 164.53.7.62 255.255.255.192
  ip broadcast-address 164.53.7.63
  ip irdp preference 90
  bridge-group 1
  bridge-group 1 path-cost 50
  bridge-group 1 input-type-list 201
!
bridge 1 protocol dec
bridge 1 priority 255
access-list 201 deny 0xFFFF 0x0000

Too Much Information

Static Routes
Static Routes

- Routes configured manually
- Useful when few or just one route exist
- Can be administrative burden
- Frequently used for default route
- Three formats:
  - Explicit next hop
  - Outbound interface
  - Both

Static Routes

- Validity of a static route is checked
  - When it is installed
  - When an interface changes its state
  - Periodically to catch dynamic changes in the routing table
- Static routes can be configured with just an address, just an interface, or both interface and address (preferred)
- The next-hop address in a static route does not have to belong to a directly connected network
  - The next-hop address just needs to be resolvable
- 0.0.0.0 used as the next-hop address is a special case
  - It is used if a static route should always follow (be resolved via) the 0.0.0.0/0 default route
Static Route Types

- **Interface only**
  ```
  ip route 10.1.0.0 255.255.0.0 Serial0
  ```

- **Next-hop only**
  ```
  ip route 10.1.0.0 255.255.0.0 20.2.2.2
  ```

- **Interface and next-hop**
  ```
  ip route 10.1.0.0 255.255.0.0 Ethernet0 20.2.2.2
  ```

- **Interface and/or next-hop with permanent keyword**
  ```
  ip route 10.1.0.0 255.255.0.0 20.2.2.2 permanent
  ```

---

Interface Only Static Routes

- Functionally equivalent to a route derived from an interface
- Are considered “directly connected” to the interface but does not come from the connected route source
- The presence of the route in the forwarding table depends on the state of the interface
- If such a static route is the direct match (not a recursive match), then IOS will use the packet’s destination address as the next-hop address
  
  This can generate a lot of ARPs on LAN segments and a lot of drops on NMBA ones (must configure map statements for each potential destination address)
Next Hop Only Static Routes

- Cisco next-hop-only routes and UNIX static routes look very much alike but behave differently.
- The next-hop address does not have to be a host address, it may be a subnet address.
- The idea is to make the static route “follow” some other route in the forwarding table.
- The next-hop address may be resolved through any route (static or dynamic) in the forwarding table, in some situations even via the default route.
- for packets being forwarded along such a route, the next-hop specified will be used as the actual next-hop address.
  - Provided that it is resolved through a directly connected route
  - Hence ARP packets are not sent for every destination address.

Next Hop and Interface Static Routes

- This technique combines Explicit association of a static route with an interface
  Explicit declaration of the next-hop to be used during the packet forwarding process
- Next-hop must resolve via interface:

```
(config)# ip route 10.0.0.0 255.0.0.0 Ethernet0/0 3.3.3.3
C    2.0.0.0/8 is directly connected, Ethernet1/0
D    3.0.0.0/8 [90/30720] via 2.2.2.3, 00:12:03, Ethernet1/0
S    10.0.0.0/8 [1/0] via 3.3.3.3, Ethernet0/0
#ping 10.10.10.6
.....
Success rate is 0 percent (0/5)
```
Static Route with Permanent Keyword

- The keyword **permanent** only changes the details of dynamic route verification
- It won’t be removed from the table even if:
  - The specified interface goes down
  - The next-hop address becomes unreachable
- However, resolvability of such route is checked:
  - When it is initially installed
  - When a “clear ip route” command is issued

Static Route Installation

- When a static route is configured, IOS checks whether a route with the same parameters has already been configured
- If the route is new, it is added to the IOS internal static route table visible via `show ip route static-table`
- The route’s resolvability is checked
  - If it is resolvable, it is installed in the forwarding table
- Installation of a static route may lead to resolvability of other static routes
  - IOS does not check for this
- Periodic static route verification handles these static routes
Redistributing Static Routes

- Redistributed via network statement
  - If next hop is interface and network specified
    ```
    ip route 172.16.1.0 255.255.255.0 ethernet 0
    router xxxx
    network 172.16.0.0
    ```
- Redistributed if so configured
  ```
  router xxxx
  redistribute static
  default metric xxxx
  ```

Floating Static Routes

- A static route with a high distance
- Can be overridden by dynamic info

```
ip route 172.16.1.0 255.255.255.0 172.16.3.1 140
router rip
network 172.16.0.0
```
Too Much Information

Default Routes

- Route used if no match is found in forwarding table
- Can be carried by routing protocols
- Two models
  - Special network number: 0.0.0.0/0
  - Flagged in routing protocol
    - IGRP uses `ip default-network`
- Can be created via `default originate`
- `default-gateway` is for ‘host mode’
Default Subnet

- Two defaults
  for unknown networks
  for unknown subnets of a known network
- Controlled by `ip classless`

RIP Example

```
router rip
network 10.0.0.0
network 172.68.0.0
redistribute static
default-metric 1
!
ip route 0.0.0.0 0.0.0.0 172.68.1.1
```

Gateway of last resort is 10.64.0.2 to network 0.0.0.0

10.0.0.0/24 is subnetted, 2 subnets
C 10.1.0.0 is directly connected, Ethernet0/0
C 10.64.0.0 is directly connected, Ethernet0/1
R* 0.0.0.0/0 [120/1] via 10.64.0.2, Ethernet0/1

172.68.0.0/24 is subnetted, 2 subnets
C 10.1.0.0 is directly connected, Ethernet0/0
C 10.64.0.0 is directly connected, Ethernet0/1
R* 0.0.0.0/0 [120/1] via 10.64.0.2, Ethernet0/1
**OSPF Example**

```
ip route 0.0.0.0 0.0.0.0 serial 0
router ospf 1
network 19.0.0.0 0.225.225.225 area 0
default-information originate always
```

- OSPF default configuration using a static route

**EIGRP Example**

```
router eigrp 1
network 192.168.0.0 0.0.255.255
network 10.1.1.1 0.0.0.0
redistribute static
default-metric 10000 1 255 1 1500
!
ip route 0.0.0.0 0.0.0.0 10.1.1.2
```

---

Copyright © 1998, Cisco Systems, Inc. All rights reserved. Printed in USA.
Presentation_ID.scr
### ISIS Example

**Diagram**

**Command**

```plaintext
ip route 0.0.0.0 0.0.0.0 s1
router isis
network 19.0.0.0
default-information originate
```

- L1 default is nearest L1L2 router
- Both L1 and L2 ISs can generate a default route
- A L1 IS will always prefer a L1 default route before using the closest L2 capable IS

### BGP Example

**Diagram**

**Command**

```plaintext
router bgp 164
default-information originate
```

- Allows redistribution of 0.0.0.0
- Same as adding network 0.0.0.0
  
  A default route must be in the table
Conditional Default

```
  ip prefix-list cond permit 10.1.1.0/24
!
  route-map def-cond permit 10
  match ip address prefix-list cond
!
  router rip
  default-information originate route-map def-cond
```

- Inserts a default route if the condition in the route map is met
- In this case, if network (prefix) 10.1.1.0/24 is present, advertise a default

Too Much Information

Route Summarization
What Is Route Summarization?

- Routing protocols can summarize addresses of several prefixes into one prefix
- This helps control resource usage

RIPv2 Summarization

- Autosummarization

```
interface Serial 0/0
ip summary-address 10.1.0.0 255.255.0.0
```

- Manual Summarization
**OSPF Inter-Area Summarization**

- Summaries sent into backbone

  ```
  B#
  router ospf 100
  area 1 range 128.213.64.0 255.255.224.0
  ```

**OSPF Inter-AS Summarization**

- Metric used is the smallest metric of all the more specific routes

  ```
  E#
  router ospf 100
  summary-address 128.213.0.0 255.255.0.0
  ```
EIGRP Summarization

- **Autosummarization**
  - 172.17.0.0/16 via NULL0

- **Manual Summarization**
  - `interface Serial 0/0`  
  - `ip summary-address eigrp 100 10.1.0.0 255.255.0.0`  
  - `10.1.0.0/16 via NULL0`  

In ISIS Summarization Is Possible...

- ...from L1 areas into the L2 backbone,  
- ...from L2 leaking down into L1 areas,  
- ...or when redistributing into L2 or L1  

```  
router isis  
summary address 192.1.0.0 255.255.0.0  
```
BGP Aggregation

• Summarization based on specifics from the BGP routing table

  aggregate-address w.x.y.z mask {as-set} {summary-only} {route-map}

• Use as-set to include path and community info from specifics

• summary-only suppresses specifics

• route-map sets other attributes

BGP Summarization Examples

• Redistribute summary 193.0.0.0

  ip route 193.0.0.0 255.0.0.0 null 0
  router bgp 100
  redistribute static

• Advertise aggregate if BGP has any more specific

  router bgp 100
  aggregate-address 193.0.0.0 255.0.0.0

• Aggregate only, more specific routes suppressed

  router bgp 100
  aggregate-address 193.0.0.0 255.0.0.0 summary-only
Summarization and Subnet Zero

- Using subnet zero can cause problems when summarization is occurring
- Router B has subnet zero of major net 168.71.0.0 configured
- Router B will advertise 168.71.0.0 as the summarized route (not the subnet zero route) to Router A because of 168.72.0.0/16
- Router A will advertise the summarized route for 168.71.5.0 to Router B
- This may cause Router A to be unable to reach 168.71.0.1
- From a routing perspective there is no difference between subnet zero and the summarized route for 168.71.0.0

Too Much Information

Filtering Route Data
Why Filter Routes?

• Prevent information about a given destination from being transmitted to neighbors
• Reduce amount of routing information within a domain
• An alternative to summarization in some situations
• Protect integrity of routing system

Filter Types

• distribute-list
• prefix-list
• route-map
**Distribute List**

access-list 10 permit 10.0.0.0 0.255.255.255

```
router rip
distribute-list 10 out
```

- Allows this router to advertise 10.0.0.0 through 10.255.255.255

```
ip access-list 100
  permit ip host 192.168.1.1 10.0.0.0 0.255.255.255

router rip
distribute-list 100 in
```

- Allows this router to accept routes about 10.0.0.0 through 10.255.255.255 from its neighbor 192.168.1.1

**Prefix List**

```
ip prefix-list foo permit 10.0.0.0/8
ip prefix-list foo deny 172.16.0.0/16 ge 24
ip prefix-list foo permit 192.168.0.0/16 le 23

router rip
distribute-list prefix foo in <gateway>
```

- Can permit or deny prefixes within a given range
- Can permit or deny destinations within a given range with a prefix length longer than specified
- Can permit or deny destinations within a given range with a prefix length shorter than specified
- Can be applied against a particular neighbor, or to all the updates received in most protocols
- Can be used to filter redistribution as well
- Simpler and easier to use than distribute lists
Route Maps

```
ip prefix-list foo permit 10.0.0.0/8
! route-map footoo permit 10
  match ip address prefix-list foo
  set metric 4
! router rip
  redistribute static route-map footoo
```

- Lots of flexibility
- Can match on many different things in the route
- Can set many different things in the route
- Useful for filtering inbound or outbound routes, or filtering redistribution

Passive Interface

```
serial 0
router xxx
  passive interface serial 0
  neighbor w.x.y.z
```

- Prevents routing updates from being transmitted out an interface
- Don’t waste resources generating updates on interfaces that have no need for them (loopback)
- Can also use passive-interface default
Example: Filtering Incoming Updates

- Control input of routing data

172.16.1.0 10.0.0.0
s0

10.0.0.0 129.1.1.0
Partner Network

distribute list 1 in serial 0
access-list 1 permit 129.1.0.0
access-list 1 deny 0.0.0.0 255.255.255.255

Example: Filtering Outgoing Updates

- Useful to propagate default route

router eigrp 111
network 128.1.0.0
distribute list 1 out serial 0
access-list 1 permit 128.1.0.0 0.0.0.0
ip default network 128.1.0.0
Precedence of Filters

- Filter routing updates in or out bound
- Interface specific or global
- Evaluation order: interface, global

Example:

```
access-list 1 deny 1.0.0.0 0.255.255.255
access-list 2 permit 1.2.3.0 0.0.0.255
router rip
distribute-list 1 in ethernet 0
distribute-list 2 in
```

- List 2 is overridden on interface ethernet 0

ACL Oversights

- Access Control Lists can filter routing updates

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Destination Port</th>
<th>Destination Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>RIP</td>
<td>UDP Port 520</td>
<td>255.255.255.255</td>
</tr>
<tr>
<td>RIPv2</td>
<td>UDP Port 520</td>
<td>224.0.0.9 (Default); 255.255.255.255</td>
</tr>
<tr>
<td>IGRP</td>
<td>IP Protocol Field 9</td>
<td>255.255.255.255</td>
</tr>
<tr>
<td>EIGRP</td>
<td>IP Protocol Field 88</td>
<td>224.0.0.10</td>
</tr>
<tr>
<td>OSPF</td>
<td>IP Protocol Field 89</td>
<td>224.0.0.5(AllOSPF Routers); 224.0.0.6(DR Routers)</td>
</tr>
<tr>
<td>ISIS</td>
<td>802 SAP 0xFEFE</td>
<td>Several Multicast MAC</td>
</tr>
<tr>
<td>BGP</td>
<td>TCP Port 179</td>
<td>Neighbor Address</td>
</tr>
</tbody>
</table>
Multi-Protocol

Running Multiple Routing Processes in the Same Box

---

The Forwarding Table

<table>
<thead>
<tr>
<th>Src</th>
<th>Network #</th>
<th>Dist/Metric</th>
<th>Next Hop</th>
<th>Age</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>D</td>
<td>198.113.181.0/24</td>
<td>170/304793</td>
<td>192.150.42.177</td>
<td>02:03:50</td>
<td>Ethernet0</td>
</tr>
<tr>
<td>O</td>
<td>198.113.178.0/26</td>
<td>110/9936</td>
<td>192.150.42.177</td>
<td>02:03:50</td>
<td>Ethernet0</td>
</tr>
<tr>
<td>R</td>
<td>192.168.96.0/24</td>
<td>120/3</td>
<td>192.150.42.177</td>
<td>00:00:20</td>
<td>Ethernet0</td>
</tr>
<tr>
<td>C</td>
<td>192.150.42.178/25</td>
<td></td>
<td>192.150.42.177</td>
<td></td>
<td>Ethernet0</td>
</tr>
</tbody>
</table>

- Populated by
  - Hardware state
  - Configuration
  - Routing protocols
The Forwarding Table

- Configuration defines what protocol processes run, which interfaces they own, and how they process protocol data
- Each routing protocol process
  - Creates its own tables and databases
  - Receives protocol packets and processes them
  - Tries to insert the results into the forwarding table

<table>
<thead>
<tr>
<th>Src</th>
<th>Network#</th>
<th>Dist/Metric</th>
<th>Next Hop</th>
<th>Age</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.12.16</td>
<td>192.168.12.16/24</td>
<td>02:03:50</td>
<td>Ethernet0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>192.168.12.17</td>
<td>192.168.12.17/24</td>
<td>02:03:50</td>
<td>Ethernet0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>192.168.12.18</td>
<td>192.168.12.18/24</td>
<td>00:00:20</td>
<td>Ethernet0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Running Multiple IGPs

- Different protocols use different metrics
- Metrics are difficult to compare algorithmically
- Therefore, a collating sequence
  - Which protocol do you believe the most?
  - Then decide which metric is the best
Use Distance to Implement Routing Policy

- Distance distinguishes sources of IP routing information

Take Route with Lowest Distance; Compare Metrics Only If Distance Is Equal

RIP Router

IGRP Router

Network A

Default Administrative Distances

<table>
<thead>
<tr>
<th>Route Source</th>
<th>Default Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connected Interface</td>
<td>0</td>
</tr>
<tr>
<td>Static Route</td>
<td>1</td>
</tr>
<tr>
<td>EIGRP Summary Route</td>
<td>5</td>
</tr>
<tr>
<td>BGP Exterior</td>
<td>20</td>
</tr>
<tr>
<td>EIGRP Internal</td>
<td>90</td>
</tr>
<tr>
<td>IGRP</td>
<td>100</td>
</tr>
<tr>
<td>OSPF</td>
<td>110</td>
</tr>
<tr>
<td>IS-IS</td>
<td>115</td>
</tr>
<tr>
<td>RIP</td>
<td>120</td>
</tr>
<tr>
<td>EGP</td>
<td>140</td>
</tr>
<tr>
<td>ODR</td>
<td>160</td>
</tr>
<tr>
<td>EIGRP External</td>
<td>170</td>
</tr>
<tr>
<td>BGP Interior</td>
<td>200</td>
</tr>
<tr>
<td>Unknown, Discard Route</td>
<td>255</td>
</tr>
</tbody>
</table>
Modifying Default Distance

- Address and mask specify the source
  \[ \text{distance weight [address mask [access-list]]} \]
- Access list applies to incoming updates
- OSPF has several based on route type:
  \[ \text{distance ospf \{[intra-area weight] [inter-area weight] [external weight]}} \]
- As does EIGRP:
  \[ \text{distance eigrp internal external} \]
- And does BGP:
  \[ \text{distance bgp external internal local} \]
- Remember the floating static route?
  \[ \text{ip route dest next-hop weight} \]

Using Distance

```
192.31.7.0
0.2

128.88.1.0

router rip
network 192.31.7.0
network 128.88.0.0
distance 225                      ! Barely believe anyone
distance 90 128.88.1.3 0.0.0.0   ! Believe the other router
distance 120 192.31.7.0 0.0.0.255 ! Default for the top net
```
Importance of Prefix Lengths

- Multiple protocols, router gets:
  - EIGRP (internal): 192.168.32.0/26
  - RIP: 192.168.32.0/24
  - OSPF: 192.68.32.0/19

- Which one goes in the table?
  - Best distance? Shortest prefix?

- They all will be!
  - They are different routes

What Is the Next Hop?

- Dest = 192.168.32.1
  - next hop = 10.1.1.1
  - falls within the 192.168.32.0/26 network
  - longest prefix 26 > 24 > 19

- Dest = 192.168.32.100
  - next hop = 10.1.1.2
  - falls within the 192.168.32.0/24
  - longest prefix 24 > 19

From previous slide:

```
router#show ip route
D 192.168.32.0/26 [90/25789217] via 10.1.1.1
R 192.168.32.0/24 [120/4] via 10.1.1.2
O 192.168.32.0/19 [110/229840] via 10.1.1.3
```
IP Classless

- Only affects the forwarding process, not the routing process
- Does not affect the way the table is built
- Without ip classless the router will not forward to supernets
- Became the default with IOS 11.3

No IP Classless

- Dest = 172.30.32.1
  next hop = 10.1.1.1
  longest prefix match
- Dest = 172.30.33.1
  next hop = 10.1.1.2
  longest prefix match
- Dest = 192.168.10.1
  next hop = 10.1.1.3
  uses default route
- Dest = 172.30.254.1
  is dropped
  unknown subnet of a known major network

```
router#show ip route
172.30.0.0/16 is variably subnetted, 2 subnets, 2 masks
D  172.30.32.0/20 [90/4879540] via 10.1.1.2
D  172.30.32.0/24 [90/25789217] via 10.1.1.1
S* 0.0.0.0/0 [1/0] via 10.1.1.3
```

Destination = 172.30.32.1
next hop = 10.1.1.1
longest prefix match

Destination = 172.30.33.1
next hop = 10.1.1.2
longest prefix match

Destination = 192.168.10.1
next hop = 10.1.1.3
uses default route

Destination = 172.30.254.1
is dropped
unknown subnet of a known major network
IP Classless

- Remote site
- No routing protocol
- Internet is reachable
- 10.0.0.0/8 is not accessible

Redistribution

Hops = bandwidth = compound = AS-PATH ?
Route Redistribution

- Router runs multiple routing protocols
- Router exchanges routes internally
- Exchange can be filtered

Redistributing Routes

- In router configuration mode, redistribute:
  - bgp | igrp | isis | ospf | static | connected | rip: the source protocol
  - metric: value for the destination protocol
  - route-map: route map for filtering
  - <xxx>: protocol specific parameters
Redistributing Routes

- Routes are redistributed from the **forwarding table**, not the protocol’s **routing table**
- Things which aren’t in the forwarding table can’t be filtered
  i.e. originating protocol for EIGRP external routes

Default Metrics

- The first, or seed, metric for a route is derived from being directly connected to a router interface
  - Re-distributed routes are not physically connected
  - “default-metric” establishes the seed metric for the route
  - Once a compatible metric is established, the metric can increment just like any other route
  - Set default metric bigger than the biggest native metric
Configuring Default Metrics

**default-metric bandwidth delay reliability loading mtu**

- Used for IGRP and Enhanced IGRP redistribution

**default-metric number**

- Used for OSPF, RIP, ISIS, and BGP redistribution

---

Example: Redistributing Routes

- RIP will distribute 192.168.1.0/24 as a connected route
- EIGRP will redistribute 192.168.1.0/24 as a RIP route

```
interface Serial0
  ip address 192.168.2.1 255.255.255.0
!
router eigrp 100
redistribute rip
default-metric 10000 1 255 1 1500
!
router rip
  network 192.168.1.0
  network 192.168.2.0
!
ip route 192.168.1.0 255.255.255.0 Serial0
```
Filtering Redistribution with Access Lists

- Filter routing updates in or out bound
- Interface specific or global or redistribution
- Evaluation order: interface, redistribution, global
- Example

```plaintext
access-list 1 deny 10.0.0.0 0.255.255.255
access-list 2 permit 10.2.3.0 0.0.0.255
router rip
default-metric 1
redistribute igrp 20
distribute-list 1 out igrp 20
distribute-list 2 in
```

Example: Route Maps for Filtering

- Redistribute RIP routes with a hop count equal to 1 into OSPF
- These routes will be redistributed into OSPF as external LSAs with
  - A metric of 5,
  - metric type of Type1
  - a tag equal to 1

```plaintext
router ospf 109
redistribute rip route-map rip-to-ospf
!
route-map rip-to-ospf permit
match metric 1
set metric 5
set metric-type type1
set tag 1
```
Redistributing Classless to Classful Protocols

- OSPF has a longer mask than RIP
- gw2 is redistributing RIP and OSPF
- RIP won’t advertise routes learned from OSPF
- Solution:

```
ip route 128.103.35.0 255.255.255.0 null0
router rip
redistribute static
default metric 1
```

Redistributing Classless to Classful Protocols

- RIP has a longer mask than OSPF
- gw2 is redistributing RIP and OSPF
- RIP won’t advertise routes learned from OSPF
- Solution:

```
ip route 128.103.35.0 255.255.255.0 null0
router rip
redistribute static
default metric 1
```
Redist Static into OSPF

- Subnet not used so:
  - 128.13.0.0 is in 16.16.16.0

- Metric not used so:
  - Metric is 20

Redist Static into OSPF

- Did not use subnet or metric keywords on redistribute static

```
interface Ethernet0
  ip address 203.250.14.2 255.255.255.0

interface Serial1
  ip address 203.250.15.1 255.255.255.252

router ospf 10
  redistribute static
  network 203.250.15.0 0.0.0.255 area 2
  network 203.250.14.0 0.0.0.255 area 0
  ip route 16.16.16.0 255.255.255.0 Ethernet0
  ip route 128.213.0.0 255.255.0.0 Ethernet0
```

```
interface Serial0
  ip address 203.250.15.2 255.255.255.252

router ospf 10
  network 203.250.15.0 0.0.0.255 area 2
```

```
sh ip route
Codes: C - connected, S - static, I - RIP, M - mobile, B - BGP
    D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
    E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
    i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
Gateway of last resort is not set

  203.250.15.0 255.255.255.252 is subnetted, 1 subnets
  C 203.250.15.0 is directly connected, Serial0
  O IA 203.250.14.0 [110/74] via 203.250.15.1, 00:02:31, Serial10
  O E2 128.213.0.0 [110/2Q] via 203.250.15.1, 00:02:32, Serial10
```
Redist Static into OSPF

**Redistribute static metric 50 subnets**

- 16.16.16.0 now appears, the cost to external routes is 50; since the external routes are of type 2 (E2), the internal cost has not been added

```text
E#sh ip route
Codes: C - connected, S - static, O - OSPF, IA - OSPF inter area
      E1 - OSPF external type 1, E2 - OSPF external type 2, i - IS-IS
      L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default

Gateway of last resort is not set

16.0.0.0 255.255.255.0 is subnetted, 1 subnets
O E2 16.16.16.0 [110/50] via 203.250.15.1, 00:00:02, Serial0
203.250.15.0 255.255.255.252 is subnetted, 1 subnets
C       203.250.15.0 is directly connected, Serial0
O IA 203.250.14.0 [110/74] via 203.250.15.1, 00:00:02, Serial0
O E2 128.213.0.0 [110/50] via 203.250.15.1, 00:00:02, Serial0
```

Redist Static into OSPF

**Redistribute static metric 50 metric-type 1 subnets**

- Note that the type has changed to E1 and the cost has been incremented by the internal cost of S0 which is 64, the total cost is 64+50=114

```text
E#sh ip route
Codes: C - connected, S - static, O - OSPF, IA - OSPF inter area
      E1 - OSPF external type 1, E2 - OSPF external type 2, i - IS-IS
      L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default

Gateway of last resort is not set

16.0.0.0 255.255.255.0 is subnetted, 1 subnets
O E1 16.16.16.0 [110/114] via 203.250.15.1, 00:04:20, Serial0
203.250.15.0 255.255.255.252 is subnetted, 1 subnets
C       203.250.15.0 is directly connected, Serial0
O IA 203.250.14.0 [110/74] via 203.250.15.1, 00:04:21, Serial0
O E1 128.213.0.0 [110/114] via 203.250.15.1, 00:04:21, Serial0
```
Redist Static into OSPF

- 128.213.0.0 permitted
- 16.16.16.0 denied

```
C# route ospf 10
    redistribute static metric 50 metric-type 1
    subnets route-map STOPUPDATE
    access-list 1 permit 128.213.0.0 0.0.255.255
    route-map STOPUPDATE permit 10
    match ip address 1
```

```
E#sh ip rou
Codes: C - connected, S - static, O - OSPF, IA - OSPF inter area
      E1 - OSPF external type 1, E2 - OSPF external type 2,
      * - candidate default

Gateway of last resort is not set

    203.250.15.0 255.255.255.252 is subnetted, 1 subnets
    C       203.250.15.0 is directly connected, Serial0
    O IA 203.250.14.0 [110/74] via 203.250.15.1, 00:00:04, Serial0
    O E1 128.213.0.0 [110/114] via 203.250.15.1, 00:00:05, Serial0
```

Feedback Loops

- When crossing a redistribution boundary, information is lost
- A physical or logical loop causes a route to be advertised back to the redistributing router that first advertised it
- How does the router know which route to accept?
  Answer: It can’t know
  Humans have to re-insert the lost information
Implementation Considerations

- Routing feedback
  - Suboptimal path selection
  - Routing loops
- Incompatible routing information
- Inconsistent convergence time

Filter to Avoid Redistribution Feedback

- Impose split horizon when redistributing
Redistribution Example

Router “Cen” Under IGRP

cen#sho ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF external type 1, N2 - OSPF external type 2, E1 - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
U - per-user static route
Gateway of last resort is not set

172.16.0.0/24 is subnetted, 11 subnets
   I  172.16.12.0 [100/1188] via 172.16.2.2, 00:00:02, TokenRing0
   I  172.16.9.0 [100/158813] via 172.16.1.1, 00:00:02, TokenRing1
   I  172.16.10.0 [100/8976] via 172.16.5.2, 00:00:02, Serial0.1
   I  172.16.11.0 [100/8976] via 172.16.4.2, 00:00:02, Serial0.2
   C  172.16.4.0 is directly connected, Serial0.2
   C  172.16.5.0 is directly connected, Serial0.1
   I  172.16.6.0 [100/160250] via 172.16.5.2, 00:00:02, Serial0.1
   I  172.16.7.0 [100/158313] via 172.16.4.2, 00:00:02, TokenRing1
   C  172.16.1.0 is directly connected, TokenRing1
   C  172.16.2.0 is directly connected, TokenRing0
   I  172.16.3.0 [100/8539] via 172.16.2.2, 00:00:02, TokenRing0
   I  172.16.4.0 [100/8539] via 172.16.1.1, 00:00:03, TokenRing2
Introduce RIP

RIP Configs

Router Cen

```
router rip
distribute igrp 1
passive-interface Serial0.2
passive-interface TokenRing0
passive-interface TokenRing1
network 172.16.0.0
default-metric 3
!
router igrp 1
distribute rip
passive-interface Serial0.1
network 172.16.0.0
default-metric 10 100 255 1 1500
```

Router R300

```
router rip
network 172.16.0.0
```
RIP Configs

Router R200

  router rip
  redistribute igrp 1
  passive-interface Serial0
  passive-interface TokenRing0
  network 172.16.0.0
  default-metric 3
!
  router igrp 1
  redistribute rip
  passive-interface Serial1
  network 172.16.0.0
  default-metric 10 100 255 1 1500

Router R100

  router rip
  network 172.16.0.0

“Cen” Doesn’t Look Too Bad

cen#sho ip route

Codes:  C - connected,  S - static,  I - IGRP,  R - RIP,  M - mobile,  B - BGP
         D - EIGRP,  EX - EIGRP external,  O - OSPF,  IA - OSPF inter area
         E1 - OSPF external type 1,  E2 - OSPF external type 2,  E - EGP
         i - IS-IS,  L1 - IS-IS level-1,  L2 - IS-IS level-2
         * - candidate default,  U - per-user static route

Gateway of last resort is not set

    172.16.0.0/24 is subnetted, 11 subnets
    I  172.16.12.0 [100/1188] via 172.16.2.2, 00:00:01, TokenRing0
    R  172.16.9.0 [120/2] via 172.16.5.2, 00:00:01, Serial0.1
    R  172.16.10.0 [120/1] via 172.16.5.2, 00:00:02, Serial0.1
    I  172.16.11.0 [100/8976] via 172.16.4.2, 00:00:02, Serial0.2
    C  172.16.4.0 is directly connected, Serial0.2
    C  172.16.5.0 is directly connected, Serial0.1
    R  172.16.6.0 [120/1] via 172.16.5.2, 00:00:02, Serial0.1
    I  172.16.7.0 [100/2688] via 172.16.1.1, 00:00:02, TokenRing1
    C  172.16.1.0 is directly connected, TokenRing1
    C  172.16.2.0 is directly connected, TokenRing0
    I  172.16.3.0 [100/8539] via 172.16.2.2, 00:00:02, TokenRing0
                  [100/8539] via 172.16.1.1, 00:00:02, TokenRing1
Not Using the Best Path

r200>sho ip route
Codes: C -- connected, S -- static, I -- IGRP, R -- RIP, M -- mobile, B -- BGP
D -- EIGRP, EX -- EIGRP external, O -- OSPF, IA -- OSPF inter area
E2 -- OSPF external type 2, E -- EGP
i -- IS-IS, L1 -- IS-IS level-1, L2 -- IS-IS level-2
* -- candidate default, U -- per-user static route

Gateway of last resort is not set

172.16.0.0/24 is subnetted, 11 subnets
I  172.16.12.0 [100/1251] via 172.16.1.2, 00:10:37, TokenRing0
I  172.16.9.0 [100/1000163] via 172.16.1.2, 00:00:37, TokenRing0
I  172.16.10.0 [100/1000163] via 172.16.1.2, 00:00:37, TokenRing0
I  172.16.11.0 [100/9039] via 172.16.1.2, 00:00:37, TokenRing0
I  172.16.4.0 [100/8539] via 172.16.1.2, 00:00:37, TokenRing0
I  172.16.5.0 [100/8539] via 172.16.1.2, 00:00:37, TokenRing0
I  172.16.6.0 [100/1000163] via 172.16.1.2, 00:00:37, TokenRing0
C  172.16.7.0 is directly connected, Serial1
C  172.16.1.0 is directly connected, TokenRing0
I  172.16.2.0 [100/751] via 172.16.1.2, 00:00:37, TokenRing0
C  172.16.1.0 is directly connected, Serial0

Use Distance to Correct

Router Cen:

router rip
redistribute igrp 1
passive-interface Serial0.2
passive-interface TokenRing0
passive-interface TokenRing1
network 172.16.0.0
default-metric 3

!
router igrp 1
redistribute rip
passive-interface Serial0.1
network 172.16.0.0
default-metric 10 100 255 1 1500
distance 130 0.0.0.0 255.255.255.255 1

! access-list 1 permit 172.16.9.0
access-list 1 permit 172.16.10.0
access-list 1 permit 172.16.6.0
Use Distance to Correct

Router R200

```conf
router rip
   redistribute igrp 1
   passive-interface Serial0
   network 172.16.0.0
   default-metric 3

router igrp 1
   redistribute rip
   passive-interface Serial1
   network 172.16.0.0
   default-metric 10 100 255 1 1500
   distance 130 0.0.0.0 255.255.255.255 1

access-list 1 permit 172.16.9.0
access-list 1 permit 172.16.10.0
access-list 1 permit 172.16.6.0
```

R200 Looks Better

```text
r200#sho ip route
Codes: C -- connected, S -- static, I -- IGRP, R -- RIP, M -- mobile, B -- BGP
D -- EIGRP, EX -- EIGRP external, O -- OSPF, IA -- OSPF inter area
E1 -- OSPF external type 1, E2 -- OSPF external type 2, E -- EGP
i -- IS-IS, L1 -- IS-IS level-1, L2 -- IS-IS level-2
* -- candidate default, U -- per-user static route
Gateway of last resort is not set

172.16.0.0/24 is subnetted, 11 subnets
I 172.16.12.0 [100/1251] via 172.16.1.2, 00:00:49, TokenRing0
R 172.16.9.0 [120/1] via 172.16.7.1, 00:00:19, Serial1
R 172.16.10.0 [120/2] via 172.16.7.1, 00:00:19, Serial1
I 172.16.11.0 [100/9039] via 172.16.1.2, 00:00:49, TokenRing0
I 172.16.4.0 [100/8539] via 172.16.1.2, 00:00:49, TokenRing0
I 172.16.5.0 [100/8539] via 172.16.1.2, 00:00:49, TokenRing0
R 172.16.6.0 [120/1] via 172.16.7.1, 00:00:19, Serial1
C 172.16.7.0 is directly connected, Serial1
C 172.16.1.0 is directly connected, TokenRing0
I 172.16.2.0 [100/751] via 172.16.1.2, 00:00:49, TokenRing0
C 172.16.3.0 is directly connected, Serial0
```
"R200’s” RIP Interface Down

Redundant Path Works

r200#sho ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2
* - candidate default, U - per-user static route

Gateway of last resort is not set

    172.16.0.0/24 is subnetted, 10 subnets
I    172.16.12.0 [100/1251] via 172.16.1.2, 00:00:08, TokenRing0
I    172.16.9.0 [130/1000163] via 172.16.1.2, 00:00:08, TokenRing0
I    172.16.10.0 [130/1000163] via 172.16.1.2, 00:00:08, TokenRing0
I    172.16.11.0 [100/9039] via 172.16.1.2, 00:00:08, TokenRing0
I    172.16.4.0 [100/8539] via 172.16.1.2, 00:00:08, TokenRing0
I    172.16.5.0 [100/8539] via 172.16.1.2, 00:00:08, TokenRing0
I    172.16.6.0 [130/1000163] via 172.16.1.2, 00:00:08, TokenRing0
C    172.16.1.0 is directly connected, TokenRing0
I    172.16.2.0 [100/751] via 172.16.1.2, 00:00:08, TokenRing0
C    172.16.3.0 is directly connected, Serial0

Copyright © 1998, Cisco Systems, Inc. All rights reserved. Printed in USA.
Presentation_ID.scr
Policy Routing

When Destinations Aren’t Enough

- Forwarding decision not based on destination address
- Selects defined path based on attributes of user packet (source/destination IP address, application port, packet lengths, and so forth)
- Set next hop or interface
- Set default next hop or interface
How Policy Routing Works

- All packets received on an interface are considered for policy routing
- Each packet is passed through a route map
- Each entry in a route map has “match” and “set” clauses
- Match clauses are conditions to be met
- If all match clauses conditions are met by the packet, then that route map entry is used and no others are considered
- An entry can be marked “permit” or “deny”
- If “deny,” normal forwarding is used
- If is “permit,” all “set” clauses are then applied and the packet is forwarded

Policy Routing Match Clauses

- Match packets against the access lists to permit policy routing of them

```
match ip address access-list-expressions
```

- If the Layer3 packet length is between min-length and max-length, inclusive, the packet matches
- Useful for distinguishing interactive versus bulk traffic when access lists will not work

```
match length min-length max-length
```
Policy Routing Set Clauses

- **set ip next-hop ip-address1 [...]**
  - Route packets to router at ip-address1

- **set ip default next-hop ip-address1 [...]**
  - If there is no explicit route for this destination, then route to this hop
  - Both use the first IP address associated with an up/up interface

Policy Routing Set Clauses

- **set interface interface1 [...]**
  - Specifies the output interface for the matched packet

- **set default interface interface1 [...]**
  - If there is no explicit route for this destination, then route to this interface
  - If interface1 is down interface2 and subsequent interfaces are tried
  - Setting interface to Null0 creates a policy that drops the packet
Policy Routing Set Clauses

**set ip precedence value**

- Set the IP precedence header field as specified
- Can use numeric or symbolic value

<table>
<thead>
<tr>
<th>Value</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Routine</td>
</tr>
<tr>
<td>1</td>
<td>Priority</td>
</tr>
<tr>
<td>2</td>
<td>Immediate</td>
</tr>
<tr>
<td>3</td>
<td>Flash</td>
</tr>
<tr>
<td>4</td>
<td>Flash-Override</td>
</tr>
<tr>
<td>5</td>
<td>Critical</td>
</tr>
<tr>
<td>6</td>
<td>Internet</td>
</tr>
<tr>
<td>7</td>
<td>Network</td>
</tr>
</tbody>
</table>

Policy Routing Configuration

- The set commands are evaluated in the following order:
  
  - `set ip precedence`
  - `set ip next-hop`
  - `set interface`
  - `set ip default next-hop`
  - `set default interface`

- A valid next hop implies the output interface
- The first combination of next hop and interface is used
Example: Policy Routing

```bash
interface Ethernet0
  ip address 192.168.93.10 255.255.255.0
  ip policy route-map foo
interface Serial1
  ip address 11.0.0.2 255.0.0.0
interface BRI0
  ip address 10.0.0.2 255.0.0.0
route-map foo permit 12
  set default interface Null0
route-map foo permit 11
  match ip address 103
  set ip next-hop 10.0.0.1
route-map foo permit 10
  match ip address 101
  set ip next-hop 11.0.0.1
access-list 101 permit tcp 192.168.93.0 0.0.0.255 any eq telnet
access-list 101 permit icmp any any
access-list 103 permit tcp 192.168.93.0 0.0.0.255 any eq ftp
```

NetFlow Policy Routing (NPR)

- Powerful traffic engineering
- ISP and/or application selection
- Distributed performance and flow acceleration
- IP precedence-based QoS
Netflow Policy Based Routing

- No flow acceleration if any match packet-size clause is used
  Packet size is not part of a flow definition
- If the router is policy routing packets to a next hop and it is down, the router will try unsuccessfully to use ARP (which is down); this behavior will continue forever
- To prevent this, configure the router to first verify that the next hop(s) of the route map is a CDP neighbor(s) before routing to that next hop
- set ip next-hop verify-availability is not supported in dCEF since it doesn’t support CDP

Example: Netflow Policy Based Routing

- Configure CEF, NetFlow, and NetFlow with flow acceleration
- Configure policy routing to verify that next hop 50.0.0.8 of route map test is a CDP neighbor before the router tries to policy route to it
- If the first packet is policy routed via route map 10, the packets of the same flow always take the same route map (10), not route map 20, because they all match or pass access list 1 check
- Policy Routing can be flow-accelerated by bypassing the access-list check

```
ip cef
ip flow-cache feature-accelerate
interface ethernet0/0/1
  ip route-cache flow
  ip policy route-map test
  route-map test permit 10
  match ip address 1
  set ip precedence priority
  set ip next-hop 50.0.0.8
  set ip next-hop verify-availability
route-map test permit 20
  match ip address 101
  set interface Ethernet0/0/3
  set ip tos max-throughput
```
Conclusion

Be Careful out There

Summary 1

• Hosts interact with routers in many ways
  Routers try to get the data delivered anyway
• Think about how much routing data is created
  Filter, summarize, use default
• IOS supports multiple routing protocols
• Under normal operation, there should be exactly
  one IGP on any network segment
  Use “passive-interface” as necessary to ensure this
Summary 2

- Run as few routing protocols as possible
  - Use static & default routes
- Each RP process has its own “Routing Table”
- IOS can exchange data between RP processes
- The number of redistribution boundaries should be kept to a minimum
- Use advanced features for special cases and for fine tuning
- Test and understand before you implement

Recommended Reading


and of course:

http://www.cisco.com
Thank You!

- Please complete and return the survey
  RST-206 Deploying IOS IP Routing Features
- Recommended sessions:
  RST-207 Deploying OSPF
  RST-208 Deploying ISIS
  RST-209 Deploying EIGRP
  RST-210 Deploying BGP4
  RST-211 ISP Architecture Essentials
  RST-410 Advanced Developments in IP Routing Technologies

Deploying Cisco IOS® IP Routing Features

Session RST-206
Please Complete Your Evaluation Form

Session RST-206