Session Goal

- To provide you with a thorough understanding of the Catalyst® 6500 switching architecture, packet flow, and key forwarding engine functions
Agenda

- Chassis Architecture
- Supervisor Engine and Switch Fabric Architecture
- Switching Module Architecture
- IPv4 Forwarding
- IP Multicast Forwarding
- Security and Feature ACLs
- QoS
- NetFlow and NetFlow Features
Catalyst 6500 Chassis Architecture

Common Features:

• Modular chassis
• Classic switching bus traces/connectors
• Crossbar fabric traces/connectors
• Redundant power supplies
• Fan tray for system cooling
  6509-NEB-A chassis offers redundant fan trays and air filtration
• Redundant voltage termination (VTT)/clock modules
• Redundant MAC address EEPROMs

6509-NEB-A chassis offers redundant fan trays and air filtration

Catalyst 6506/6509 Chassis Architecture

Catalyst 6506/6509 Chassis

• Slots 1 and 2—Supervisor 1 or 2, or switching module
• Slots 5 and 6—Supervisor 720, SFM/SFM2, or switching module
• Other slots—Any switching module
• 2 fabric channels per slot

NEB/NEB-A Chassis Have Vertical Slot Alignment

© 2004 Cisco Systems, Inc. All rights reserved. Printed in USA.
Presentation_ID.scr
Catalyst 6513 Chassis Architecture

Catalyst 6513 Chassis

- Slots 1 and 2—Supervisor 2 or switching module
- Slots 7 and 8—Supervisor 720, SFM2, or switching module
- 1 fabric channel slots 1–8
  Dual-fabric modules not supported in slots 1–8!
- 2 fabric channels slots 9–13
  Any switching module

Catalyst 6503 Chassis Architecture

Catalyst 6503 Chassis

- Slots 1 and 2—Supervisor engine, or switching module
- Other slots—Any switching module
- 2 fabric channels per slot
- Power supplies in rear
  Power entry module (PEM) provides power connection

SFM/SFM2 and CEF720 Modules Not Supported in This Chassis!
Power Management

- Supervisor, switching modules, daughter cards, and Powered Devices (PDs) all require power
  - Power allocation predetermined based on Part Number
- Use the power calculator on cisco.com to determine power requirements and minimum power supply
  http://www.cisco.com/go/powercalculator
- If insufficient power available, system powers down PDs, then switching modules, then services modules
  - PDs and modules powered off from highest to lowest (port or slot)

White Paper on Power over Ethernet on Catalyst 6500:

Agenda

- Chassis Architecture
- Supervisor Engine and Switch Fabric Architecture
- Switching Module Architecture
- IPv4 Forwarding
- IP Multicast Forwarding
- Security and Feature ACLs
- QoS
- NetFlow and NetFlow Features
Supervisor Engine 2

- Integrated PFC2 daughter card
- Integrated 300MHz R7000 Switch Processor CPU
- Optional MSFC2 daughter card with 300MHz R7000 Route Processor CPU
- Supports optional Switch Fabric Module (SFM)/SFM2
- Supports one external PCMCIA flash slot
- 2 x GbE GBIC uplink ports

Supported from Cisco IOS 12.1(5c)EX and Catalyst OS 6.1(1)/12.1(3a)E1
Supervisor Engine 2/PFC2 Architecture

- QoS TCAM contains QoS ACL entries
- FIB contains IPv4 prefix entries
- ACL CAM contains security and feature ACL entries
- Supervisor 2 Baseboard
  - ADJ contains rewrite info
  - NetFlow table for stats and features
- MSFC2 Daughter Card
  - DRAM
  - RP (MSFC3) CPU
  - SP (NMP) CPU
- Port ASIC
  - GbE Uplinks
- Replication engine for multicast/SPAN
- Interface to fabric and bus
- L2 CAM contains MAC entries
- 16 Gbps Bus
- DBUS, RBUS
- LCDBUS, LCRBUS

Supervisor Engine 720

- Integrated 720Gbps fabric
- Integrated PFC3 daughter card
- Integrated 600MHz SR71000 RP/SP CPUs on MSFC3 daughter card
- Provides dual external compact flash slots
- 2 x GbE uplink ports—
  - 2 x SFP <or>
  - 1 x SFP and 1 x 10/100/1000

Supported in Cisco IOS 12.2(14)SX and Catalyst OS 8.1(1) with 12.2(14)SX2
Crossbar Switch Fabric

- Provides multiple conflict-free paths between switching modules
  - Dedicated bandwidth per slot
- 18 fabric channels in total
- Two fabric channels per slot in 6503/6506/6509
- In 6513:
  - One fabric channel slots 1–8
  - Two fabric channels slots 9–13
  - “Dual-fabric” modules not supported in slots 1–8 of 6513
Switch Fabric Module and SFM2

- 256 Gbps crossbar switch fabric
- Works with Supervisor 2 and CEF256/dCEF256 modules
- Fabric channels run at 8 Gbps full duplex
  8 Gbps in/8 Gbps out per channel
- Fabric module occupies separate slot
  - 6506/6509—Slots 5 and 6
  - 6513—Slots 7 and 8
- SFM—Supports 6506 and 6509
- SFM2—Supports 6506, 6509, and 6513
- Not supported in 6503

Supervisor 720 Switch Fabric

- 720 Gbps crossbar switch fabric
- Integrated on Supervisor 720 baseboard
- Works with all fabric-capable modules
  - Fabric channels auto-sync speed on per-slot basis (8 Gbps or 20Gbps)
- Fabric channels run at 20 Gbps full duplex
  - 20 Gbps in/20 Gbps out per channel
- Different slot requirements for Supervisor 720 in 6506/6509/6513:
  - 6506/6509—Supervisor goes in slot 5 or 6
  - 6513—Supervisor goes in slot 7 or 8
Monitoring Fabric Status and Utilization

- Cisco IOS: `show fabric [active | channel-counters | errors | fpoe | medusa | status | switching-mode | utilization]`
- Catalyst OS: `show fabric {channel {counters | switchmode | utilization} | status}`

```
6506# show fabric utilization
     slot  channel  speed Ingress %  Egress %
       1      0      8G     22          23
       2      0      8G      4          9
       3      0     20G      0          1
       3      1     20G     11         12
       4      0     20G      0          1
       4      1     20G     10         13
       6      0     20G      0          1
6506#
```

Policy Feature Cards

- Daughter card for supervisor engine
- Provides the key components enabling high-performance hardware packet processing
- Supervisor 2 supports PFC2
- Supervisor 720 supports:
  - PFC3A
  - PFC3BXL
Policy Feature Cards (Cont.)

Key Hardware-Enabled Features:
• Layer 2 switching
• IPv4 unicast forwarding
• IPv4 multicast forwarding
• Security ACLs
• QoS/policing
• NetFlow statistics

PFC3 Also Supports:
• IPv6, MPLS, Bidir PIM, NAT/PAT, GRE/v6 tunnels

PFC Comparison

<table>
<thead>
<tr>
<th>Feature</th>
<th>PFC2</th>
<th>PFC3A</th>
<th>PFC3BXL</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIB TCAM</td>
<td>256K</td>
<td>256K</td>
<td>1M</td>
</tr>
<tr>
<td>Adjacency Table</td>
<td>256K</td>
<td>1M</td>
<td>1M</td>
</tr>
<tr>
<td>NetFlow Table</td>
<td>128K (32K)</td>
<td>128K (64K)</td>
<td>256K (230K)</td>
</tr>
<tr>
<td>MAC Table</td>
<td>128K (32K)</td>
<td>64K (32K)</td>
<td>64K (32K)</td>
</tr>
<tr>
<td>IPv6</td>
<td>Software</td>
<td>Hardware</td>
<td>Hardware</td>
</tr>
<tr>
<td>Bidir PIM</td>
<td>Software</td>
<td>Hardware</td>
<td>Hardware</td>
</tr>
<tr>
<td>Native MPLS</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>EoMPLS</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>NAT, Tunnels</td>
<td>Software</td>
<td>Hardware</td>
<td>Hardware</td>
</tr>
<tr>
<td>uRPF Check</td>
<td>Yes (Single Path)</td>
<td>Yes (Multipath)</td>
<td>Yes (Multipath)</td>
</tr>
<tr>
<td>IPX</td>
<td>Hardware (Hybrid)</td>
<td>Software</td>
<td>Software</td>
</tr>
</tbody>
</table>
## Agenda

- Chassis Architecture
- Supervisor Engine and Switch Fabric Architecture
- Switching Module Architecture
- IPv4 Forwarding
- IP Multicast Forwarding
- Security and Feature ACLs
- QoS
- NetFlow and NetFlow Features

### PFC Comparison (Cont.)

<table>
<thead>
<tr>
<th>Feature</th>
<th>PFC2</th>
<th>PFC3A</th>
<th>PFC3BXL</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACL TCAM</td>
<td>32K/4K</td>
<td>32K/4K Dual-Bank</td>
<td>32K/4K Dual-Bank</td>
</tr>
<tr>
<td>PACLs</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>ACL Counters</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>QoS TCAM</td>
<td>32K/4K</td>
<td>32K/4K</td>
<td>32K/4K</td>
</tr>
<tr>
<td>ACL Labels</td>
<td>512</td>
<td>512</td>
<td>4K</td>
</tr>
<tr>
<td>ACL LOUs</td>
<td>32</td>
<td>64</td>
<td>64</td>
</tr>
<tr>
<td>User-Based Policing</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Egress Policing</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>HSRP/VRRP Groups</td>
<td>16/system</td>
<td>No limit</td>
<td>No limit</td>
</tr>
<tr>
<td>Unique MAC/Interface</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Classic Module Architecture

Classic Module

Port ASIC

Port ASIC

Port ASIC

Port ASIC

Port ASIC for physical connectivity, buffering, and queueing
CEF256 Module Architecture

Fabric interface to interface with fabric and bus
Replication engine for local SPAN/multicast replication
Port ASIC for physical connectivity, buffering, and queueing

CEF720 Module Architecture

Fabric interface/replication engine to interface with fabric and bus, and for multicast/SPAN replication
Layer 2 Engine for FIB/Adj, ACL, QoS and NetFlow lookups
Layer 2 Engine for L2 lookups
Port ASIC for physical connectivity, buffering, and queueing
Distributed Forwarding

- One or more modules have local forwarding engine (DFC—Distributed Forwarding Card)
- Central engine and distributed engines perform different lookups independently and simultaneously
- Implementation is fully distributed
  - All the hardware from corresponding PFC is present on the DFC
  - Full Layer 2, FIB, Adjacency, ACL/QoS information downloaded from Supervisor
  - Ingress DFC performs all lookups locally
- Deterministic, highly scalable—Not flow-based
- NOT just for local switching—destination interface irrelevant
- DFCs always require Cisco IOS software and a switch fabric

DFC/DFC3A

- DFC works in conjunction with specific supervisor
  - DFC works with PFC2 on Supervisor 2
  - DFC3A works with PFC3 on Supervisor 720
- DFC is optional daughter card for CEF256 modules
  - WS-F6K-DFC=
- DFC3 is optional daughter card for CEF256/CEF720 modules
  - WS-F6K-DFC3A = for CEF256 modules
  - WS-F6700-DFC3A = for CEF720 modules
- WS-X6816-GBIC module REQUIRES either DFC or DFC3
- Local CPU for managing hardware tables
- Use remote login module command to access DFC console
  - Commands available on DFC console are for troubleshooting use only, under direction from Cisco TAC/Escalation
Agenda

• Chassis Architecture
• Supervisor Engine and Switch Fabric Architecture
• Switching Module Architecture
• IPv4 Forwarding
• IP Multicast Forwarding
• Security and Feature ACLs
• QoS
• NetFlow and NetFlow Features

IPv4 FORWARDING
Hardware CEF-Based Forwarding

- Catalyst 6500 leverages existing software Cisco Express Forwarding (CEF) model
- Supervisor 2/PFC2 and Supervisor 720/PFC3 extend CEF to hardware
- What is Cisco Express Forwarding (CEF), in a nutshell?
  - Boil down the routing table = FIB table
  - Boil down the ARP table = adjacency table
- FIB table contains IP prefixes
- Adjacency table contains next-hop information

Hardware CEF-Based Forwarding (Cont.)

- Decouples control plane and data plane
  - Forwarding tables built on control plane
  - Tables downloaded to hardware for data plane forwarding
- Hardware CEF process:
  - FIB lookup based on destination prefix (longest-match)
  - FIB “hit” returns adjacency, adjacency contains rewrite information (next-hop)
  - ACL, QoS, and NetFlow lookups occur in parallel and effect final result
Hardware IPv4 Unicast Entries

- IPv4 Lookup — 10.1.1.1
- 72 bits
- 10.1.1.100
- MASK (/32)
- ...
- 10.1.3.0
- 10.1.2.0
- MASK (/24)
- ...
- 10.1.0.0
- 172.16.0.0
- MASK (/16)
- ...
- 0.0.0.0
- MASK (/0)

- Result Memory
- RPF VLANs, ADJ Pointer
- Load-Balancing Hash

- Source IP
- Dest IP
- Optional L4 ports
- Unique ID

- RW MACs, VLAN, Encap

---

FIB TCAM and Adjacencies

- Overall FIB TCAM entries shared by
  - IPv4
  - IPv4 multicast
  - IPv6
  - MPLS

- Hardware adjacency table also shared
- Actual adjacency table entries are NOT shared
Displaying IPv4 Forwarding Summary Information

- **Cisco IOS:**
  - `show mls cef summary`
  - `show mls cef statistics`
  - `show mls statistics`
  - `show mls cef hardware`

- **Catalyst OS:**
  - `show mls cef`
  - `show mls`

```
6509-neb#show mls cef summary
Total routes: 8309
  IPv4 unicast routes: 5948
  IPv4 multicast routes: 2359
  MPLS routes: 0
  IPv6 unicast routes: 0
  IPv6 multicast routes: 0
  EoM routes: 0
```

```
6509-neb#
```

Displaying Hardware IPv4 Prefix Entries

```
6509-neb#show mls cef
Codes: decap - Decapsulation, + - Push Label

<table>
<thead>
<tr>
<th>Index</th>
<th>Prefix</th>
<th>Adjacency</th>
</tr>
</thead>
<tbody>
<tr>
<td>64</td>
<td>127.0.0.51/32</td>
<td>receive</td>
</tr>
<tr>
<td>65</td>
<td>127.0.0.0/32</td>
<td>receive</td>
</tr>
<tr>
<td>66</td>
<td>127.255.255.255/32</td>
<td>receive</td>
</tr>
<tr>
<td>67</td>
<td>0.0.0.0/32</td>
<td>receive</td>
</tr>
<tr>
<td>68</td>
<td>255.255.255.255/32</td>
<td>receive</td>
</tr>
<tr>
<td>75</td>
<td>10.10.1.1/32</td>
<td>receive</td>
</tr>
<tr>
<td>76</td>
<td>10.10.1.0/32</td>
<td>receive</td>
</tr>
<tr>
<td>77</td>
<td>10.10.1.255/32</td>
<td>receive</td>
</tr>
<tr>
<td>78</td>
<td>10.10.1.2/32</td>
<td>Gi1/1,</td>
</tr>
<tr>
<td>3200</td>
<td>224.0.0.0/24</td>
<td>receive</td>
</tr>
<tr>
<td>3201</td>
<td>10.10.1.0/24</td>
<td>glean</td>
</tr>
<tr>
<td>3202</td>
<td>10.100.0.0/24</td>
<td>Gi1/1,</td>
</tr>
<tr>
<td>3203</td>
<td>10.100.1.0/24</td>
<td>Gi1/1,</td>
</tr>
<tr>
<td>3204</td>
<td>10.100.2.0/24</td>
<td>Gi1/1,</td>
</tr>
<tr>
<td>3205</td>
<td>10.100.3.0/24</td>
<td>Gi1/1,</td>
</tr>
</tbody>
</table>
```

- **Cisco IOS:** `show mls cef`
- **Catalyst OS:** `show mls entry cef ip`
Displaying Detailed Hardware IPv4 Prefix and Adjacency Entries

- **Cisco IOS:**
  - `show mls cef <prefix> [detail]`
  - `show mls cef adjacency [entry <entry> [detail]]`

- **Catalyst OS:**
  - `show mls entry cef ip <prefix/mask> [adjacency]`

```
6509-neb# show mls cef 10.100.20.0 detail
<>
M(3222) : E | 1 FFF 0 0 0 0   255.255.255.0
V(3222) : 8 | 1 0    0 0 0 0   10.100.20.0   (A:98304, P:1, D:0, m:0, B:0)

6509-neb# show mls cef adjacency entry 98304
Index: 98304
smac: 000f.2340.5dc0, dmac: 0000.0000.0013
mtu: 1518, vlan: 1019, dindex: 0x0, l3rw_vld: 1
packets: 4203, bytes: 268992
```

Finding the Longest-Match Hardware Prefix Entry

- **Cisco IOS:** `show mls cef lookup <ip_address> [detail]`

```
6509-neb# show mls cef lookup 10.101.1.0
Codes: decap - Decapsulation, + - Push Label
Index  Prefix              Adjacency
6509-neb# show mls cef lookup 10.101.1.0
Codes: decap - Decapsulation, + - Push Label
Index  Prefix              Adjacency
3203   10.101.0.0/16       Gi2/12,     0007.b30a.8bfc
```

© 2004 Cisco Systems, Inc. All rights reserved. Printed in USA.
Presentation_ID.scr
IPv4 CEF Load Sharing

- Up to 8 hardware load-sharing paths per prefix
- Use `maximum-paths` command in routing protocols to control number of LB paths
- IPv4 CEF load-balancing is per-IP flow
- Per-packet load-balancing NOT supported
- Load-sharing based on Source and Destination IP addresses by default
- Configuration option supports inclusion of L4 ports in the hash (mls ip cef load-sharing full)
- “Unique ID” in Supervisor 720 prevents polarization (can be changed with ip cef load-sharing algorithm universal command)

Example of Hardware Load-Balancing Prefix Entry

- `show mls cef`
- `show mls cef lookup`

```
6509-neb#show mls cef lookup 10.100.20.1

Codes: decap - Decapsulation, + - Push Label
Index  Prefix               Adjacency
3222   10.100.20.0/24     Gi1/1, 0000.0000.0013
       Gi1/2, 0000.0000.0014
       Gi2/1, 0000.0373.e078
       Gi2/2, 0000.0373.e079

6509-neb#
```
Calculating Which Load-Balancing Path Traffic Will Follow

show mls cef exact-route

6509-neb#show mls cef exact-route 10.77.17.8 10.100.20.199
   Interface: Gi1/1, Next Hop: 10.10.1.2, Vlan: 1019, Destination Mac: 0000.0000.0013

6509-neb#show mls cef exact-route 10.44.91.111 10.100.20.199
   Interface: Gi2/2, Next Hop: 10.40.1.2, Vlan: 1018, Destination Mac: 0000.0373.e079

IPv4 Unicast RPF Check

10.255.0.0/16
  10.20.0.0/16

System Supports Only ONE uRPF Mode—Strict or Loose! Last Configured Mode Overrides!
Reference: Classic to Classic Centralized Forwarding

1. Unicast IPv4 packet received on Classic Module A; entire packet is flooded on DBUS and all devices, including the PFC on the supervisor engine, receive it
2. PFC makes a forwarding decision for the packet
3. PFC floods forwarding decision result on RBUS
4. Egress port ASIC on Classic Module B is selected to transmit the packet—all other devices on the bus discard the packet
Reference: CEF256 to CEF256 Centralized Forwarding

1. Unicast IPv4 packet received on CEF256 Module A; entire packet is flooded on LCDBUS and fabric interface receives it
2. Fabric interface floods just the packet header on the DBUS; PFC receives packet header and makes a forwarding decision for the packet
3. PFC floods forwarding decision result on RBUS
4. Fabric interface transmits packet across the fabric
5. CEF256 Module B receives the packet and transmits the packet, and the result, on its LCDBUS; the egress port ASIC is selected to transmit the packet
Reference: CEF720/DFC3 to CEF720/DFC3 Distributed Forwarding

1. Unicast IPv4 packet received on CEF720 Module A; entire packet is forwarded to the fabric interface
2. Fabric interface sends just the packet header to the DFC; DFC makes a forwarding decision for the packet
3. DFC returns the forwarding decision result to the fabric interface
4. Fabric interface transmits packet across the fabric
5. CEF720 Module B receives the packet and transmits the packet to the egress port ASIC
Agenda

- Chassis Architecture
- Supervisor Engine and Switch Fabric Architecture
- Switching Module Architecture
- IPv4 Forwarding
- IP Multicast Forwarding
- Security and Feature ACLs
- QoS
- NetFlow and NetFlow Features
Hardware IPv4 Multicast Overview

- Support for central and distributed IPv4 multicast hardware switching
- Off-load forwarding tasks from RP CPU
  RP only handles control plane functions (run multicast protocols, maintain state)
- SP CPU also performs some control plane functions
  IGMP snooping
  Managing hardware forwarding entries
- Supports (S,G) and (*,G) in hardware
- Supervisor 720 also supports BiDir (*,G) in hardware
- Supports distributed multicast replication

Multicast Forwarding Tables

- RP CPU derives 3 key data structures from multicast routing table
  Multicast FIB—Consists of (S,G) and (*,G) entries, and RPF VLAN
  Adjacency table—Contains rewrite info and MET index
  Multicast Expansion Table (MET)—Contains output interface lists (OILs), i.e., lists of interfaces requiring replication
- RP CPU downloads tables to SP CPU
- SP CPU installs tables in the appropriate hardware
  Multicast FIB and adjacency tables installed in PFC/DFC hardware
  MET installed in replication engines
- SP CPU also maintains L2 table for IGMP snooping
IGMP Snooping

- **Purpose**—Constrains multicast flooding on Layer 2 ports
- Implementation across Catalyst switch products very similar
  
PFC ASICs recognize IGMP packets and redirect them to SP CPU ("protocol redirection logic")
  
Switch installs static Layer 2 forwarding entries for each multicast group MAC
  
Multicast data traffic forwarded to appropriate interfaces according to MAC address table entries (per VLAN)
  
- **Does not affect performance for multicast data traffic**
  
Protocol redirection ONLY redirects IGMP packets, not UDP (data) packets

---

Supervisor Engine 720 Multicast Architecture

- FIB contains (S,G) and (.*G) entries, and RPF VLAN
- ADJ contains rewrite info and MET index
- MET contains OILs for replication
- SP CPU programs hardware, performs IGMP snooping
- RP CPU maintains state, downloads tables
- CPU Daughter Card
  - RP (MSFC3) CPU
- SP (NMP) CPU
- Replication engine replicates packets to OILs
- L2 CAM contains multicast MAC entries
  - Protocol redirection captures IGMP packets
- L2 CAM contains multicast MAC entries
CEF256 Module Multicast Architecture

MET contains OILs for replication

Replication engine replicates packets to OILs

CEF720 Module Multicast Architecture

MET contains OILs for replication

Replication engine replicates packets to OILs

L2 CAM and protocol redirection logic
Hardware IPv4 Multicast Entries

Displaying Summary Hardware Multicast Information

- Cisco IOS:
  - show mls ip multicast summary
  - show mls ip multicast statistics

- Catalyst OS:
  - show mls multicast
  - show mls multicast statistics

6506#show mls ip multicast summary
21210 MMLS entries using 3394656 bytes of memory
Number of partial hardware-switched flows: 0
Number of complete hardware-switched flows: 21210

Directly connected subnet entry install is enabled
Hardware shortcuts for mvpn mroutes supported
Current mode of replication is Ingress
Auto-detection of replication mode is enabled
Consistency checker is enabled
Bidir gm-scan-interval: 10
6506#
IP Mroute Table with Complete Shortcut

- show ip mroute

6506#show ip mroute 10.3.1.100 239.1.1.100

(10.3.1.100, 239.1.1.100), 00:01:33/00:02:52, flags: T
Incoming interface: GigabitEthernet3/1, RPF nbr 0.0.0.0, RPF-MFD
Outgoing interface list:
  GigabitEthernet4/2, Forward/Sparse-Dense, 00:00:19/00:02:41, H
  GigabitEthernet4/1, Forward/Sparse-Dense, 00:00:19/00:02:49, H
  GigabitEthernet3/2, Forward/Sparse-Dense, 00:00:19/00:02:47, H
  Vlan200, Forward/Sparse-Dense, 00:01:14/00:01:48, H
  Vlan150, Forward/Sparse-Dense, 00:01:14/00:01:46, H
  Vlan100, Forward/Sparse-Dense, 00:01:15/00:01:54, H

6506#

IP Mroute Table with Partial Shortcut

- show ip mroute

6506#show ip mroute 10.3.1.100 239.1.1.100

(10.3.1.100, 239.1.1.100), 00:07:17/00:02:58, flags: T
Incoming interface: GigabitEthernet3/1, RPF nbr 0.0.0.0, Partial-SC
Outgoing interface list:
  Vlan100, Forward/Sparse-Dense, 00:01:13/00:02:42, H
  Vlan150, Forward/Sparse-Dense, 00:01:16/00:01:43, H (ttl-threshold 64)
  Vlan200, Forward/Sparse-Dense, 00:01:16/00:01:43, H
  GigabitEthernet4/2, Forward/Sparse-Dense, 00:06:03/00:02:35, H
  GigabitEthernet4/1, Forward/Sparse-Dense, 00:06:04/00:02:43, H
  GigabitEthernet3/2, Forward/Sparse-Dense, 00:06:04/00:02:37, H

6506#
Displaying Hardware Multicast Forwarding Entries

- Cisco IOS: `show mls ip multicast`
- Catalyst OS: `show mls multicast entry`

```
6506# show mls ip multicast
Multicast hardware switched flows:
(10.3.1.100, 239.1.1.100) Incoming interface: Gi3/1, Packets switched: 720396460
Hardware switched outgoing interfaces:
  Gi3/2 Vlan100 Vlan150 Gi4/1 Gi4/2 Vlan200
  RPF-MFD installed

(10.3.1.103, 230.100.1.1) Incoming interface: Gi3/1, Packets switched: 443201
Hardware switched outgoing interfaces:
  Gi3/2 Gi4/1
  RPF-MFD installed
<-->
```

CEF720/DFC3 to CEF720/DFC3 Multicast Distributed Forwarding (1)
Reference: CEF720/DFC3 Multicast Distributed Forwarding Packet Flow

1. IP multicast packet is received on Module A from Source in Blue VLAN and is sent to the fabric interface/replication engine (FI/RE) ASIC
2. FI/RE ASIC sends the packet header to the local DFC3; the DFC3 forwarding engine receives the packet header and performs the packet lookup
3. DFC3 sends the lookup result to the FI/RE
4. FI/RE sends the packet to the port ASIC with a receiver in the Blue VLAN; the port ASIC bridges the packet to the receiver
5. The FI/RE on Module A also generates a copy of the packet on a special internal VLAN (Black) based on the MET; the FI/RE sends the packet headers to the local DFC3; the DFC3 forwarding engine receives the packet header and performs the packet lookup
6. DFC3 sends the lookup result to the FI/RE
7. The result indicates the packet must be bridged across the fabric to Module B; the FI/RE on Module B receives the packet on the special internal VLAN
8. The FI/RE sends the packet headers to the local DFC3 forwarding engine for a packet lookup
9. DFC3 sends the lookup result to the FI/RE
10. The FI/RE on Module B generates a copy of the packet for each local OIF (Orange, Green, Red, and Purple VLANs) based on the MET; the FI/RE sends the packet headers to the local DFC3; the DFC3 forwarding engine receives the packet headers and performs the packet lookup for each packet
11. DFC3 sends the lookup result for each packet lookup to the FI/RE
12. The FI/RE on Module B forwards the packets to the appropriate ports with receivers attached

Agenda

• Chassis Architecture
• Supervisor Engine and Switch Fabric Architecture
• Switching Module Architecture
• IPv4 Forwarding
• IP Multicast Forwarding
• Security and Feature ACLs
• QoS
• NetFlow and NetFlow Features
Security ACLs

- Enforce security policies based on Layer 3 and Layer 4 information

- Three varieties:
  - Router ACLs (RACLs) — IPv4, IPX*, IPv6**
  - VLAN ACLs (VACLs) — IPv4, IPX*, MAC
  - Port ACLs (PACLs)** — IPv4, MAC

- Dedicated ACL TCAM ensures security ACLs do not affect other system functions

* IPX ACLs in Supervisor 2 Only!
** IPv6 in Hardware on Supervisor 720 Only!
*** PACLs in Supervisor 720 Only!
**Feature ACLs**

- Override FIB forwarding decision to allow alternative processing
- Feature ACLs used for:
  - Policy-Based Routing (PBR)
  - Reflexive ACLs
  - Network Address Translation (NAT/PAT)
- Typically paired with NetFlow table and/or Adjacency table
- Sophisticated feature merge algorithm allows multiple security and feature ACLs to be applied to a single interface/VLAN

---

**ACL Merge**

- What is merging?
  - May need two or more ACL features on a single interface (e.g., RACL and PBR)
  - Hardware supports limited number of ACL lookups on a single packet
  - Merge produces ACEs that return correct result in a single lookup
- Downside: Can cause TCAM blowup
  - ACE intersection/interrelations can require lots of TCAM entries
- Two algorithms: ODM and BDD
- Nutshell: USE ODM whenever possible!
- Supervisor 720 dual-bank TCAM architecture may avoid merge entirely

White Paper on ACL Merge Algorithms and ACL Hardware Resources:
### Monitoring ACL TCAM Utilization

```
6509-neb# show tcam counts

<table>
<thead>
<tr>
<th></th>
<th>Used</th>
<th>Free</th>
<th>Percent Used</th>
<th>Reserved</th>
</tr>
</thead>
<tbody>
<tr>
<td>Labels:</td>
<td>23</td>
<td>4073</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>ACL_TCAM</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Masks:</td>
<td>2902</td>
<td>1194</td>
<td>70</td>
<td>72</td>
</tr>
<tr>
<td>Entries:</td>
<td>15261</td>
<td>17507</td>
<td>46</td>
<td>576</td>
</tr>
<tr>
<td>QOS_TCAM</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Masks:</td>
<td>7</td>
<td>4089</td>
<td>0</td>
<td>18</td>
</tr>
<tr>
<td>Entries:</td>
<td>32</td>
<td>32736</td>
<td>0</td>
<td>144</td>
</tr>
<tr>
<td>LOU:</td>
<td>47</td>
<td>81</td>
<td>36</td>
<td></td>
</tr>
<tr>
<td>ANDOR:</td>
<td>1</td>
<td>15</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>ORAND:</td>
<td>0</td>
<td>16</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>ADJ:</td>
<td>0</td>
<td>2048</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

6509-neb#```

- **Cisco IOS:** `show tcam counts`
- **Catalyst OS:** `show security acl resource-usage`

### Verifying Hardware ACL Enforcement

```
6509-neb# show fm summary

6509-neb# show fm summary
Interface: Vlan199 is up
TCAM screening for features: ACTIVE inbound
Interface: Vlan400 is up
TCAM screening for features: ACTIVE inbound
TCAM screening for features: ACTIVE outbound
Interface: Vlan402 is up
TCAM screening for features: ACTIVE inbound
TCAM screening for features: ACTIVE outbound
Interface: Vlan404 is up
TCAM screening for features: ACTIVE inbound
TCAM screening for features: ACTIVE outbound
Interface: Vlan405 is up
TCAM screening for features: ACTIVE inbound

6509-neb#

fm = “Feature Manager”
ACTIVE = ACL Policy Is Installed in Hardware```

© 2004 Cisco Systems, Inc. All rights reserved. Printed in USA.
Presentation_ID.scr
Displaying Hardware ACL “Hit Counters”

Cisco IOS: `show tcam interface <interface> acl {in | out} ip`

```
6509-neb# show tcam interface vlan199 acl in ip
<>
  permit udp any 10.89.210.0 0.0.0.255 (234265 matches)
  permit udp any 10.90.143.0 0.0.0.255 (6860 matches)
  permit udp any 10.91.25.0 0.0.0.255 (23 matches)
  permit udp any 10.92.82.0 0.0.0.255 (23662 matches)
  permit udp any 10.93.154.0 0.0.0.255 (3232 matches)
  permit udp any 10.94.1.0 0.0.0.255 (12113 matches)
  permit udp any 10.95.109.0 0.0.0.255 (247878 matches)
  permit udp any 10.96.201.0 0.0.0.255 (33234 matches)
  permit udp any 10.97.16.0 0.0.0.255 (6855 matches)
  permit udp any 10.98.43.0 0.0.0.255 (89745 matches)
  permit udp any 10.1.1.0 0.0.0.255 (7893485 matches)
deny ip any any (448691555 matches)
6509-neb#
```

ACL Hit Counters Supported on PFC3BXL Only!

---

**Catalyst 6500 Security ACL Comparison**

<table>
<thead>
<tr>
<th>Feature</th>
<th>Supervisor 2</th>
<th>Supervisor 720</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPv4 RACLs</td>
<td>Yes (32K)</td>
<td>Yes (32K)</td>
</tr>
<tr>
<td>IPv4/MAC VACLs</td>
<td>Yes (32K)</td>
<td>Yes (32K)</td>
</tr>
<tr>
<td>IPv4 Reflexive ACLs</td>
<td>Yes (128K)</td>
<td>Yes (128K/256K*)</td>
</tr>
<tr>
<td>IPv6 RACLs</td>
<td>Software</td>
<td>Hardware (8K)</td>
</tr>
<tr>
<td>IPv6 Reflexive ACLs</td>
<td>Software</td>
<td>Hardware (64K/128K)*</td>
</tr>
<tr>
<td>PACLS</td>
<td>No</td>
<td>Yes (32K)</td>
</tr>
<tr>
<td>ACL TCAM Entries/Masks</td>
<td>32K/4K</td>
<td>32K/4K</td>
</tr>
<tr>
<td>Dual-Bank TCAM</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>ACL Hit Counters</td>
<td>No</td>
<td>Yes*</td>
</tr>
<tr>
<td>ACL Labels</td>
<td>512</td>
<td>512/4094*</td>
</tr>
<tr>
<td>LOUs</td>
<td>32</td>
<td>64</td>
</tr>
</tbody>
</table>

* PFC3B-XL

* = Change from Earlier Version
Agenda

- Chassis Architecture
- Supervisor Engine and Switch Fabric Architecture
- Switching Module Architecture
- IPv4 Forwarding
- IP Multicast Forwarding
- Security and Feature ACLs
- QoS
- NetFlow and NetFlow Features
**Catalyst 6500 QoS Model**

---

**Input Queue Scheduling**

- Input scheduling only performed if port configured to trust COS
- Scheduling based on input COS
- Implements tail-drop thresholds
  - Thresholds at which packets with different COS values are dropped
- Queue structure example: 1p1q4t
  - One strict-priority queue, one standard queue with four tail-drop thresholds

FAQ: What Are The Buffer Sizes and Queue Structures for the Different Modules?
Input Queue Scheduling Details

1p1q4t

- Threshold 4 (COS 6,7) — 100%
- Threshold 3 (COS 4) — 75%
- Threshold 2 (COS 1,2,3) — 60%
- Threshold 1 (COS 0) — 50%

Tail-drop thresholds — If queue depth greater than configured threshold, additional received packets associated with that threshold are dropped.

Classification

- Selects traffic for further QoS processing
  - Marking
  - Policing
- Based on —
  - Port trust
  - QoS ACLs
QoS ACLs

- Used to classify traffic based on Layer 3 and Layer 4 information
- Hardware support for standard and extended IPv4 and MAC QoS ACLs
- Use QoS TCAM and other ACL resources to classify traffic for marking and policing
- Dedicated QoS TCAM
  32K entries/4K masks
- Share other resources (LOUs and labels) with security ACLs

Marking

- Untrusted port—Set a default QoS value
- Trusted port—Use the marking (COS, precedence, DSCP) provided by upstream device
- QoS ACLs—Set QoS values based on standard or extended ACL match
Policing

- Defines a policy for traffic on a port or VLAN, based on the rate at which traffic is received
- Based on a classic token bucket scheme
  - Tokens (1 byte each) added to bucket at fixed rate (up to max)
  - Packets with adequate tokens are “in profile”: packet transmitted, tokens removed from bucket
  - Packets without adequate tokens are dropped or marked down
- Note! PFC2 uses Layer 3 packet size; PFC3 uses Layer 2 frame size

Policing Details

- **Aggregate policers**—Bandwidth limit applied cumulatively to all flows that match the ACL
  - Example—All FTP flows limited in aggregate to configured rate
- **Microflow policers**—Bandwidth limit applied separately to each individual flow that matches the ACL
  - Example—Each individual FTP flow limited to configured rate
  - Leverages NetFlow table
- Supervisor 2 and Supervisor 720 support **INGRESS** policing, on a per-switchport, per-Layer 3 interface, or per-VLAN basis
- Supervisor 720 also supports **EGRESS** aggregate policing on a per-VLAN or per-Layer 3 interface basis
Monitoring Service Policies (Marking and Policing)

```
6506# show policy-map interface vlan 100
Service-policy input: VLAN-100
  class-map: NET-44-TCP (match-all)
    Match: access-group name POL-44-TCP
      100000000 bps 100000 limit 100000 extended limit
    Earl in slot 6:
      2940073472 bytes
      5 minute offered rate 358172704 bps
      aggregate-forwarded 608631808 bytes action: transmit
      exceeded 2331441664 bytes action: drop
      aggregate-forward 100352000 bps exceed 384495616 bps
  class-map: NET-55 (match-all)
    Match: access-group name MARK-55
    set precedence 5:
    Earl in slot 6:
      2940069888 bytes
      5 minute offered rate 358172616 bps
      aggregate-forwarded 2940069888 bytes
```

- Cisco IOS: `show policy interface`
- Catalyst OS: `show qos statistics {aggregate-policer | l3stats}`

Policing (Rate Limiting) vs. Shaping

<table>
<thead>
<tr>
<th>Traffic Rate</th>
<th>Time</th>
<th>Policing</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Traffic Rate</th>
<th>Time</th>
<th>Shaping</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Reclassification/Remarking

- Policing action may reclassify and remark certain traffic
  
  For example, transmit with marked-down DSCP

Congestion Avoidance

Weighted Random Early Detection (WRED):

- Congestion AVOIDANCE mechanism
- **Weighted** because some classes of traffic are more important or sensitive than others
- **Random** in that the packets to discard are randomly chosen within a class
  
  Which classes are more subject to discards is configurable
- Prevents global TCP window synchronization and other disruptions
WRED Thresholds

• Each queue has multiple WRED thresholds
• **Low threshold** is the point at which random discards will begin for a particular class
• **High threshold** is the point at which tail-drop for the particular class begins
• As buffers fill…
  
  Rate of discards increases for traffic associated with lower thresholds
  Higher thresholds are reached, and new traffic classes are subject to random discards

WRED Operation (1)

• Two classes, two thresholds each:
  
  **Gold**
  - 100% high
  - 60% low

  **Blue**
  - 80% high
  - 30% low

• When queue depth exceeds 30%, some random blue packets are dropped
WRED Operation (2)

- Two classes, two thresholds each:
  - Gold: 100% high, 60% low
  - Blue: 80% high, 30% low

- As queue depth increases, drop rate for blue packets increases

WRED Operation (3)

- Two classes, two thresholds each:
  - Gold: 100% high, 60% low
  - Blue: 80% high, 30% low

- When queue depth exceeds 60%, drop rate for blue packets increases and gold packets become subject to random drops
WRED Operation (4)

- Two classes, two thresholds each:
  - Gold:
    - 100% high
    - 60% low
  - Blue:
    - 80% high
    - 30% low
- When queue depth exceeds 80%, tail-drop occurs for blue packets (all exceed packets dropped), and drop rate for gold packets increases.

Output Queue Scheduling

- Scheduling based on **COS**
- Implements **tail-drop or WRED thresholds**
- Queue structure example: **1p3q8t**
  - One strict-priority queue, three standard queues with eight WRED thresholds each
Output Queue Scheduling Operation

Switch Fabric

(D)WRR Used to Schedule Between Normal Queues

Strict Priority Queue Serviced First if Traffic Present

Strict

1p3q8t

Egress Port

Weights (Expressed as Ratio) Determine How Much Traffic Is Transmitted from Each Queue

WRR and DWRR Scheduling

- **Weighted Round Robin (WRR)**
  - Uses ratio to determine number of packets to transmit from one queue before moving to the next queue
  - Higher weight = more packets transmitted from that queue
  - Unfair with variable-length packets in different queues

- **Deficit WRR**
  - Also uses ratio, but tracks bytes in each queue using deficit counter
  - Packet(s) transmitted during queue servicing only if size of next packet to transmit is <= deficit counter
  - Deficit counter “refreshed” at beginning of each queue servicing period
  - Results in fair scheduling over time
Monitoring Ingress and Egress Queuing

- Cisco IOS: `show queuing interface`
- Catalyst OS: `show qos statistics <mod/port>`

```
6506#show queuing interface gig 1/2 | begin Packets dropped
Packets dropped on Transmit:
   BPDU packets: 0
   queue thresh dropped [cos-map]
   -----------------------------------------------
   1   1      5994368 [0 1 ]
   1   2          8 [2 3 ]
   2   1        3444 [4 6 ]
   2   2    0* [7 ]
   3   1    0* [5 ]

* - shared transmit counter
```

QoS Action Points—CEF256 to CEF256 Centralized Forwarding

[Diagram of QoS Action Points—CEF256 to CEF256 Centralized Forwarding]
**QoS Action Points—CEF720/DFC3 to CEF720/DFC3 Distributed Forwarding**

**Agenda**

- Chassis Architecture
- Supervisor Engine and Switch Fabric Architecture
- Switching Module Architecture
- IPv4 Forwarding
- IP Multicast Forwarding
- Security and Feature ACLs
- QoS
- NetFlow and NetFlow Features
NETFLOW AND NETFLOW FEATURES

IPv4 NetFlow

- Used to track statistics for traffic flows through the system
- IPv4 statistics entries created in NetFlow table when new flows start
- Entries removed when flows expire
  Timer and session based expiration
- Flow statistics can be exported using NetFlow Data Export (NDE)
- Theoretical maximum utilization versus effective utilization
  Varies based on hardware implementation and hash efficiency
Displaying NetFlow Statistics Entries

- Cisco IOS: `show mls netflow ip`
- Catalyst OS: `show mls statistics entry`

```
6506#show mls netflow ip
Displaying Netflow entries in Supervisor Earl
DatIP          SrcIP          Prot:SrcPort:DstPort  Src i/f   :AdjPtr
---------------------------------------------------------------- ------------
Pkts        Bytes        Age  LastSeen  Attributes
---------------------------------------------------
10.102.130.213  10.214.39.79    tcp :46528  :www     :0x0
7            3766          17    15:47:37   L3 - Dynamic
10.230.215.148  10.155.22.221   tcp :51813  :45912   :0x0
25           21329         47    15:47:39   L3 - Dynamic
10.97.36.200    10.17.64.177    tcp :65211  :www     :0x0
9            7664          17    15:47:38   L3 - Dynamic
10.90.33.185    10.46.13.211    tcp :27077  :60425   :0x0
10           5734          17    15:47:38   L3 - Dynamic
```

Supervisor 2 NetFlow Table

- PFC2 NetFlow table contains 128K entries
- Entries installed in NetFlow table via hash algorithm
- Lookup key based on IP header information
  - Which information depends on flow mask
  - Values fed into hash function to generate lookup key
  - Key identifies row in NetFlow table containing flow information
- Hash ~25% efficient (32K entries)
  - 17-bit hash key used
  - Probability of collision increases after 32K entries
- NetFlow lookup results:
  - Hit—Update statistics for existing flow
  - Miss—Create new NetFlow table entry
  - Hash collision—Move to next page
  - All pages full—No statistics for flow
Reference: Supervisor 2 NetFlow Processing

1. Layer 3 and Layer 4 information (based on flow mask) extracted from packet header to generate NetFlow lookup key

2. NetFlow lookup key passed to hash function

3. Hash function generates 17-bit hash key identifying correct NetFlow table row

4. Lookup key compared to contents of identified row on 1st page; no match, so move through pages and compare

5. Lookup key matches contents of table row on 4th page; statistics for flow updated
Supervisor 720 NetFlow Table

- PFC3 NetFlow table size varies
  PFC3A—128K entries
  PFC3B-XL—256K entries
- Entries installed in NetFlow table via TCAM-assisted hash algorithm
- Lookup key based on IP header information
  - Which information depends on flow mask
  - Values fed into hash function to generate lookup key
  - Key entry stored in NetFlow TCAM
  - TCAM hit returns NetFlow table index
  - NetFlow table contains actual flow information
- Hash ~50–90% efficient (64/230K entries)
  - 36-bit hash key used
  - Probability of overflow increases after 64K/230K entries

Supervisor 720 NetFlow Table (Cont.)

- Hash key and NetFlow index stored in TCAM
  - Two banks of 64K/128K rows of 36 bit entries for keys
- NetFlow table arranged as 1 page with 128/256K rows
- Alias CAM (128 entries) handles unlikely case of hash collision
- NetFlow lookup results:
  - Hit—Update statistics for existing flow
  - Miss—Create new NetFlow table entry
  - Hash collision—Create alias CAM entry
  - Alias CAM full—No statistics for flow
Reference: Supervisor 720 NetFlow Processing

1. Layer 3 and Layer 4 information (based on flow mask) extracted from packet header to generate NetFlow lookup key
2. NetFlow lookup key passed to hash function, generating 36-bit hash key
3. Hash key looked up in NetFlow TCAM
4. Match in NetFlow TCAM identifies correct NetFlow table index
5. Lookup key compared to contents of location in NetFlow table
6. Lookup key matches contents of location in NetFlow table; statistics for flow updated
Monitoring NetFlow Table Usage and Creation Failures

- Cisco IOS: `show mls netflow table-contention`
- Catalyst OS: `show mls debug`

```
6506# show mls netflow table-contention detailed
Earl in Module 6
Detailed Netflow CAM (TCAM and ICAM) Utilization
===============================================
TCAM Utilization : 100%
ICAM Utilization : 0%
Netflow TCAM count : 13944
Netflow ICAM count : 0
Netflow Creation Failures : 270274
Netflow CAM aliases : 0
```

NetFlow Aging

- Determining when to remove existing NetFlow entries from the table
- Three types of aging
  - Normal—Fixed idle time for flows
  - Fast—Threshold-based aging of flows
  - Long—Maximum lifetime for flows
- Also have session-based entry removal
- Default timers are conservative
  - Tuning is recommended!
  - More aggressive normal aging timer
  - Enable fast aging
Changing and Viewing the NetFlow Aging Configuration

- **Cisco IOS:**
  
  mls aging {normal | fast | long}
  
  show mls netflow aging

- **Catalyst OS:**
  
  set mls agingtime [fast | long-duration]
  
  show mls

```bash
6506# show mls netflow aging

<table>
<thead>
<tr>
<th>enable</th>
<th>timeout</th>
<th>packet threshold</th>
</tr>
</thead>
<tbody>
<tr>
<td>normal aging true</td>
<td>300</td>
<td>N/A</td>
</tr>
<tr>
<td>fast aging false</td>
<td>32</td>
<td>100</td>
</tr>
<tr>
<td>long aging true</td>
<td>1920</td>
<td>N/A</td>
</tr>
</tbody>
</table>

6506#
```
Related Networkers Sessions

- RST-2504—Cisco Catalyst 6500 Service Module Design and Implementation
- RST-2505—Campus Design Fundamentals
- RST-2506—Analyzing the Impact of Emerging Technologies on Campus Design
- RST-2514—High Availability in Campus Network Deployments
- RST-3509—Troubleshooting Cisco Catalyst 6500 Series Switches
- RST-3511—Troubleshooting LAN Protocols

Complete Your Online Session Evaluation!

**WHAT:** Complete an online session evaluation and your name will be entered into a daily drawing

**WHY:** Win fabulous prizes! Give us your feedback!

**WHERE:** Go to the Internet stations located throughout the Convention Center

**HOW:** Winners will be posted on the onsite Networkers Website; four winners per day