SEC-2101:

Network Core Infrastructure Protection: Best Practices

Alvaro Retana (aretana@cisco.com)

Technical Leader, IP Routing Deployment and Architecture

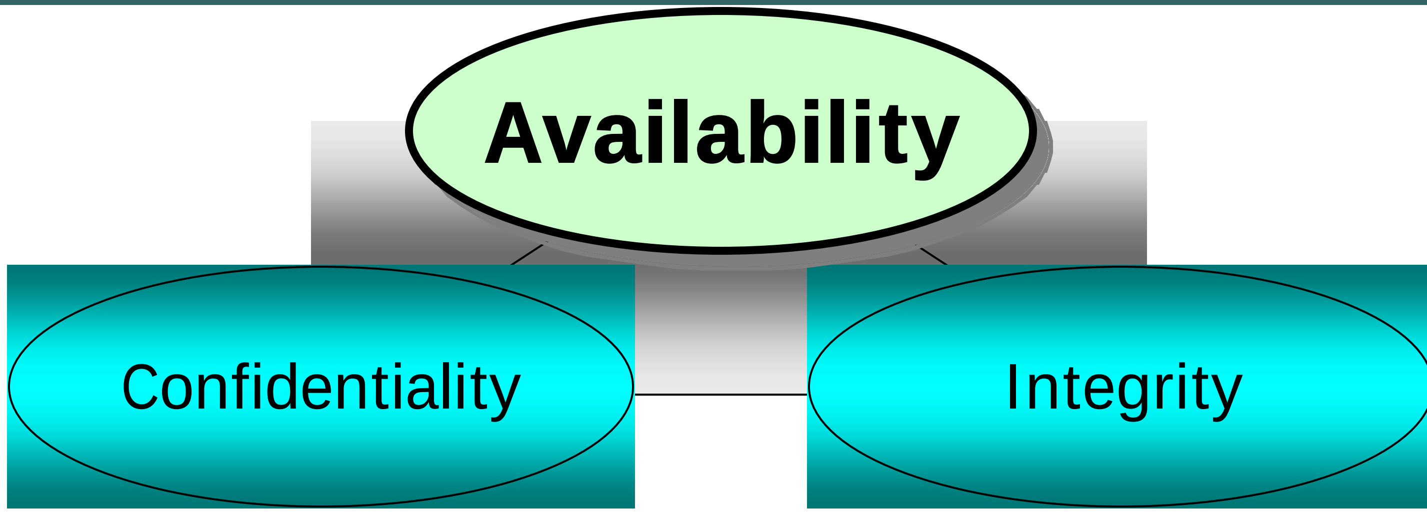
Recuerde siempre:

- Apagar su teléfono móvil/pager, o usar el modo "silencioso".
- Completar la evaluación de esta sesión y entregarla a los asistentes de sala.
- Ser puntual para asistir a todas las actividades de entrenamiento, almuerzos y eventos sociales para un desarrollo óptimo de la agenda.
- Completar la evaluación general incluida en su mochila y entregarla el miércoles 8 de Junio en los mostradores de registración. Al entregarla recibirá un regalo recordatorio del evento.

Agenda

- Infrastructure security overview
- Preparing The Network
- Router Security: A Plane Perspective
- Tools and Techniques
- Platform Architecture
- Conclusions

The Security Trinity



- –**Confidentiality**
- –**Integrity**
- –**Availability**

Network Availability: Protect the Infrastructure

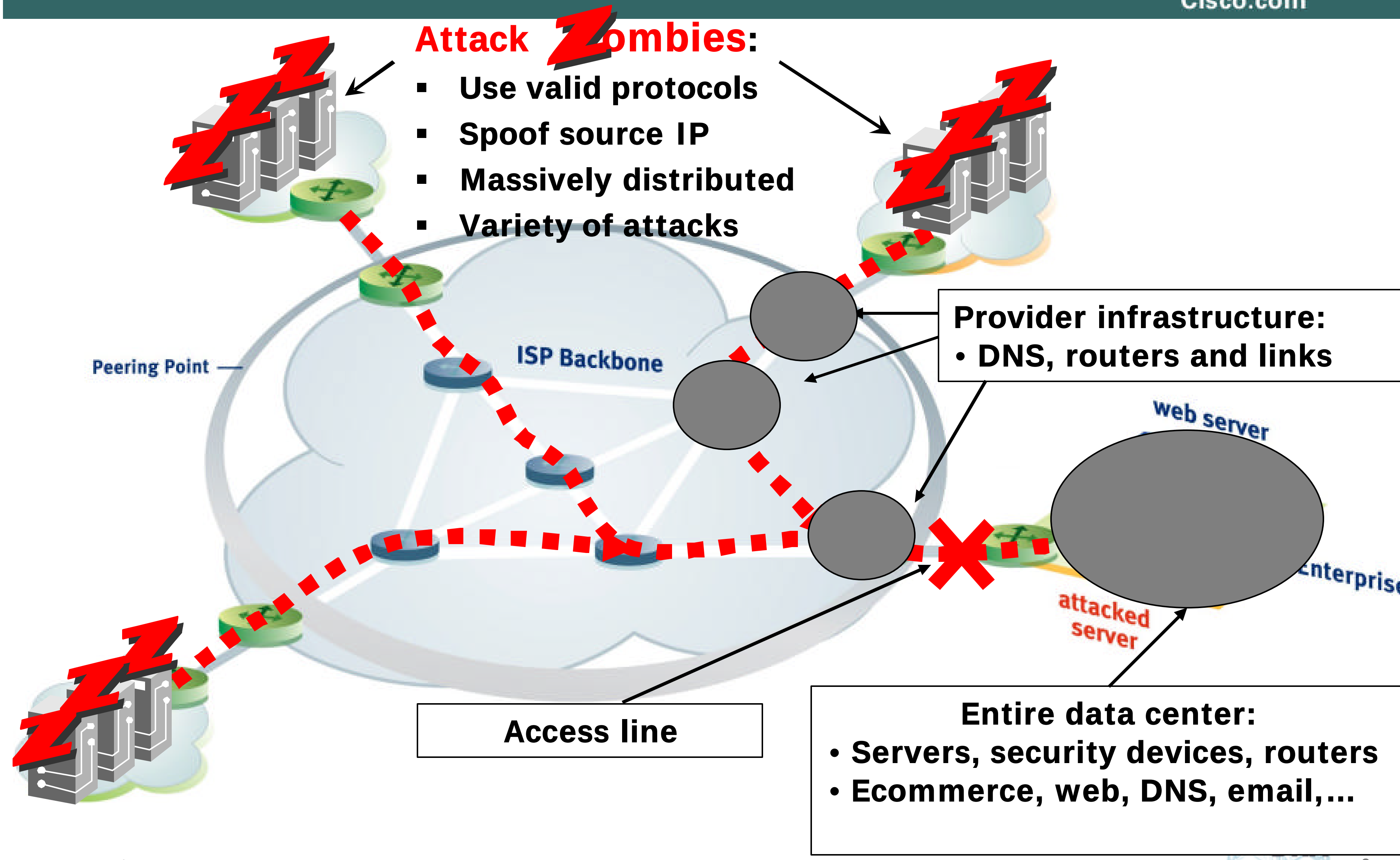
Cisco.com

- **We have a multitude of end device security products and technologies but the core is critical**
- **Remember: availability**

  Protecting the infrastructure is the most fundamental security requirement
- **Infrastructure protection should be included in all disaster recovery and high availability designs**

  Part of network design
- **Without an available core, no services (e.g. voice) can be delivered**

DDoS Vulnerabilities
Multiple Threats and Targets



Denial of Service Trends

Cisco.com

- **Multi-path**
  - Truly distributed
    - Routeservers, large botnets
- **Multi-vector**
  - SYN AND UDP AND…
- **Increased use "state"**
  - Looks like valid traffic (e.g. http get)
  - Can consume resources at various levels of the network
- **Financial incentive**
  - SPAM, DoS-for-hire
  - Large, thriving business
  - Forces us to reassess the risk profile

Infrastructure Attacks

- The infrastructure is no longer a “black box”

    Sites with Cisco documents and presentations on routing protocols (and I don’t mean Cisco.com)

    Marked increase in presentations about routers, routing and IOS vulnerabilities at conferences like Blackhat, Defcon and Hivercon

    Router attack tools and training are being published

- Why mount high-traffic DDOS attacks when you can take out your target's gateway routers?
- Hijacked routers are valuable in the spam world, which has a profit driver
- Router compromise (0wn3d) due to weak password

From Bad to Worms

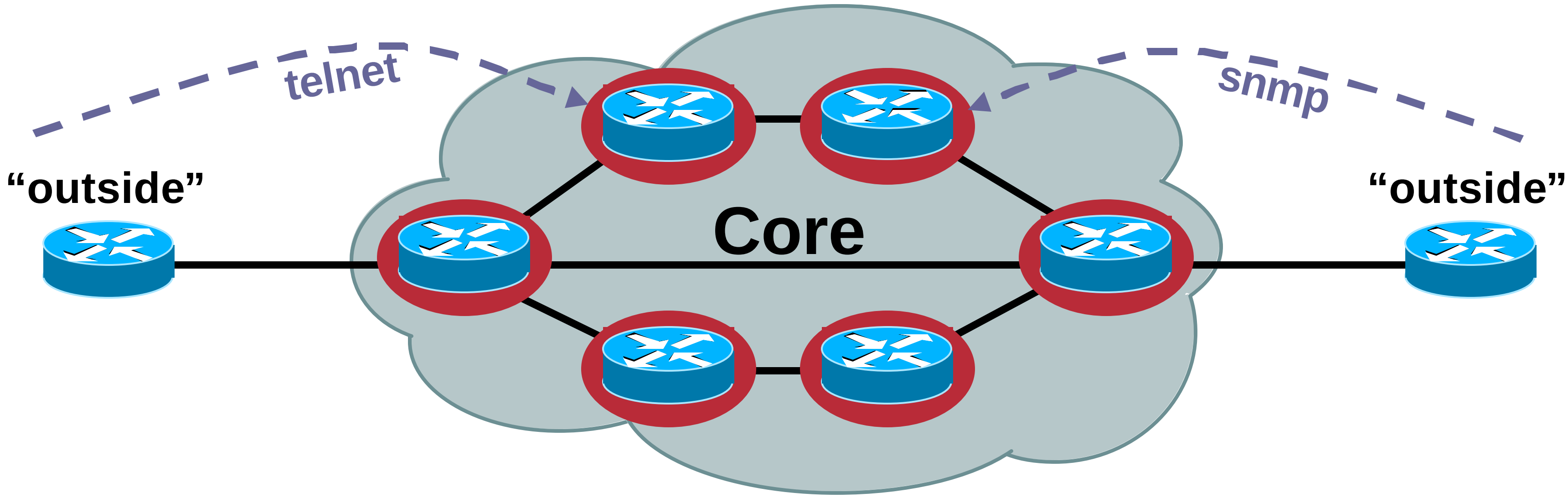
- Worms have emerged as the new security reality
- Old worms never die!
  - Millions of CodeRed(!) and Slammer packets still captured daily
- Most worms are intended to compromise hosts
- Worm propagation is dependant on network availability
- Worms and DoS are closely related
  - Secondary worm effects can lead to denial of service
  - Worms enable DoS by compromising hosts → BOTnets
- Perimeters are crumbling under the worm onslaught (VPN/mobile workers, partners, etc.)
- Don’t neglect viruses!

Worms and the Infrastructure

- Worms typically infect end-stations
- To date, worms have not targeted infrastructure BUT secondary effects have wreaked havoc
  - Increased traffic
  - Random scanning for destination
  - Destination address is multicast
  - Header variances
- At the core SP level, the aggregate effects of a worm can be substantial

The Old World: Network Edge

Cisco.com

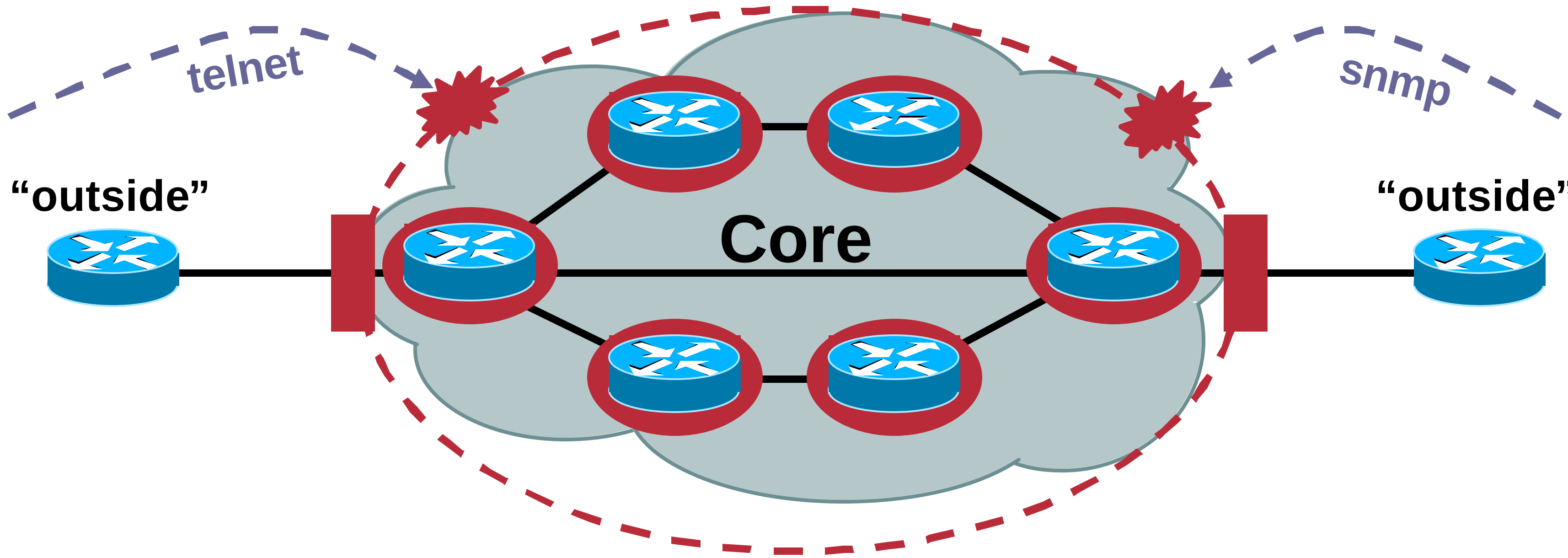


- Core routers individually secured
- Every router accessible from outside

Routing DNA

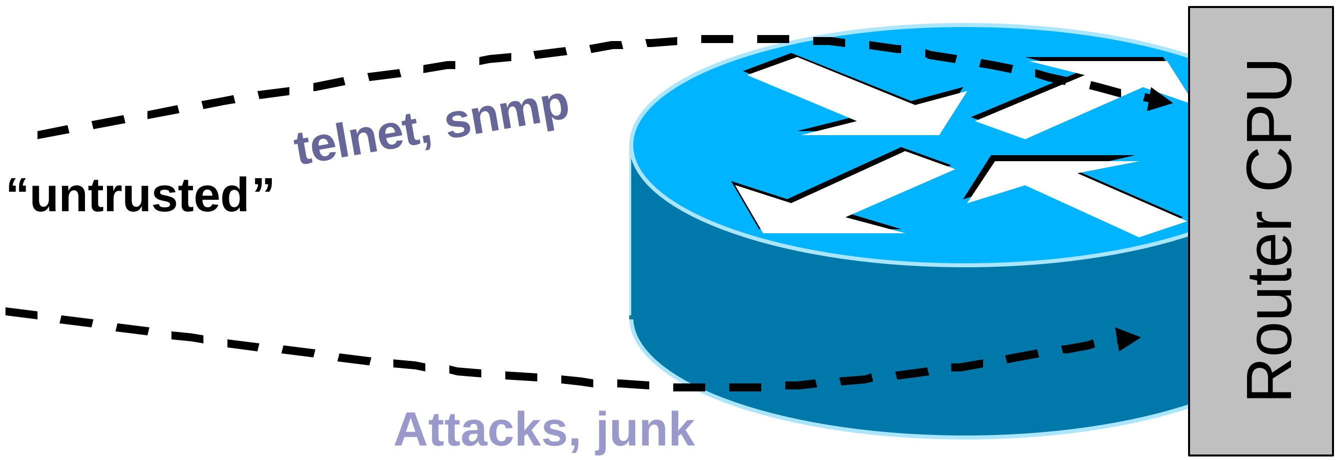
The New World: Network Edge

Cisco.com



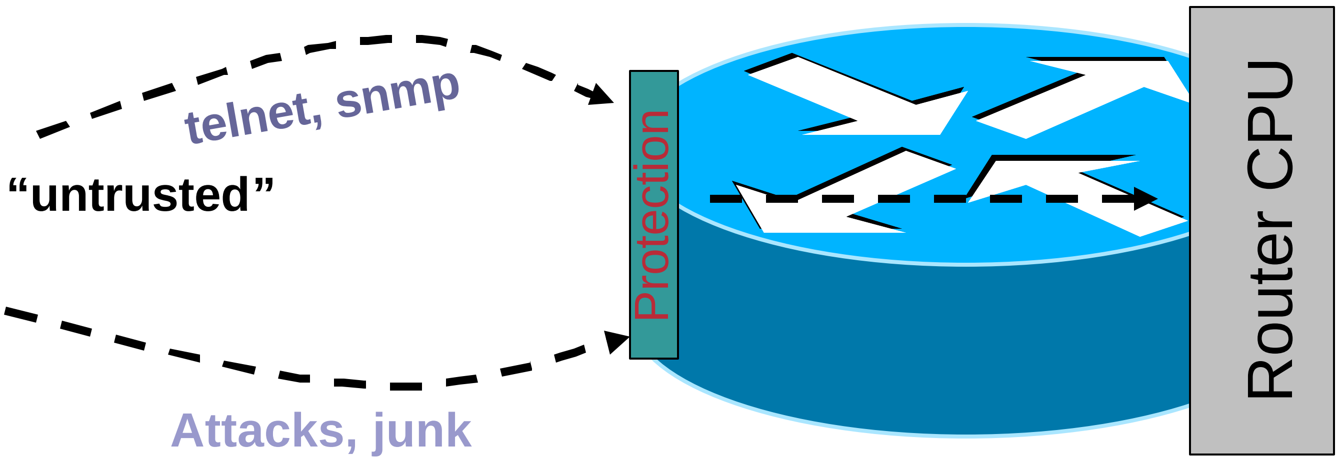
- Core routers individually secured PLUS
- Infrastructure protection
- Routers generally NOT accessible from outside

The Old World: Router Perspective



- Policy enforced at process level (VTY ACL, SNMP ACL, etc.)
- Some early features such as ingress ACL used when possible

The New World: Router Perspective



- Central policy enforcement, prior to process level
- Granular protection schemes
- On high-end platforms, hardware implementations

Agenda

- Infrastructure security overview
- Preparing The Network
- Router Security: A Plane Perspective
- Tools and Techniques
- Platform Architecture
- Conclusions

Preparing The Network

- This is a whole topic onto itself

    Best practices can help prevent infection

    Attack mitigation is rarely effective without best practice deployment

- "I want to stop the DoS but I haven't implemented XYZ yet" or "I don't know who to contact"
- Best practices can be tough to deploy, but the benefits are immeasurable

Preparing The Network

- **Limit Attack Vectors**
  - **Traffic filtering both incoming AND outgoing connections**
  - **Source address validation (ACL and/or uRPF)**
  - **RFC2827 filtering where applicable**
  - **BGP policy enforcement**
- **Identify/Detect Attacks**
  - **Develop network baseline, including traffic analysis**
  - **Logging and log analysis**
  - **IDS at strategic locations**

Preparing The Network

- **Periodic security scans of internal network to identify policy violations**
- **Ongoing security vulnerability awareness**
- **Routine security auditing**
- **Event monitoring and correlation for firewalls, IDS, network devices and servers**

Infrastructure Specific Protection Techniques

- **Protect the infrastructure itself from attack**
  - **From the Inside – Users/Customers**
  - **From the Outside – Peers/Upstreams**
- **Methodology:**
  - **Erect an edge barrier (infrastructure ACLs)**
  - **Focus on the device specific configuration (receive ACL and control plane policing)**
  - **Understand the platform architecture and how it impacts security**

Infrastructure Best Practices

- **Harden Routers and Switches**
  - **Secure management access**
  - **Secure routing protocols**
  - **Develop and deploy standard configs that reflect security policy**
    - **Leverage configuration tools like RANCID**
  - **Understand the technology (e.g. VLAN security principles)**
  - **Understand the architecture, performance characteristics and features of the devices**

DEVICE HARDENING

Disable Unneeded Services

- no service finger
- no service udp-small-servers
- no service tcp-small-servers
- no ip http server
- no ip redirects
- no ip directed-broadcast
- no ip proxy-arp

Cisco Discovery Protocol

- **CDP can be used to learn information about neighboring devices that are running CDP**
  - **IP address, software version…**
- **CDP is configured per interface**
- **Disable CDP when it isn't needed**
  - **Public facing interfaces**

Source Routing / IP Options

- IP has a provision to allow source IP host to specify route through Internet
- ISPs should turn this off, unless it is specifically required:

```
no ip source-route
```

- Packets with IP Options can be dropped or the options can be ignored (12.0(23)S / 12.3(4)T):

```
ip options drop
ip options ignore
```

ICMP Unreachable Overload

- **Packets that cannot be forwarded are punted for ICMP Unreachable generation.**
- **Risk → high number of unreachables overloading CPU**

  no ip unreachables

- **All Routers with any static route to Null0 should put *no ip unreachables***
- **If Unreachables are needed, use ICMP Unreachable Rate-Limiting Command:**

  ip icmp rate-limit unreachable [DF] <1-4294967295 millisecond>

  no ip icmp rate-limit unreachable [df]

  Default is 500 milliseconds

What Ports Are Open on the Router?

- It may be useful to see what sockets/ports are open on the router
- ***Show ip sockets***—show some of the UDP ports opened

```
IOSRouter#show ip sockets
ProtoRemote      Port      Local       Port  In Out Stat TTY
OutputIF
 17 192.190.224.195   162 204.178.123.178  2168   0   0    0   0
 17   --listen--          204.178.123.178    67   0   0    9   0
 17 0.0.0.0           123 204.178.123.178   123   0   0    1   0

 17 0.0.0.0             0 204.178.123.178   161   0   0    1   0
```

What Ports Are Open on the Router?

Cisco.com

- **Two steps required for TCP ports:**

  show tcp brief all

  show tcp tcb

```
GSR-1#sh tcp bri all
TCB       Local Address            Foreign Address          (state)
52F6D218  60.20.1.2.11002          60.20.1.1.179            ESTAB
52F7065C  50.20.1.1.179            50.20.1.2.11007          ESTAB
52F6CD8C  *.*                      *.*                      LISTEN
537D0944  *.179                    60.20.1.1.*              LISTEN
537CE2C4  *.179                    50.20.1.2.*              LISTEN
```


Routing DNA

Network Time Protocol

- **Synchronize time across all devices**
- **When security event occurs, data will have consistent timestamps**
    - **From external time source:**
        - **Upstream ISP, Internet, GPS, atomic clock**
    - **From internal time source**
    - **Router can act as stratum 1 time source**

```
ntp source loopback0
ntp server 10.223.1.1 source loopback0
ntp authenticate
ntp authentication-key number md5 value
…
```

Configuring Syslog on a Router

- **Syslog data is invaluable**
  - **Attack forensics**
  - **Day to day events and debugging**
- **To log messages to a syslog server host, use the logging global configuration command**
  - **`logging`** ***`host`***
  - **`logging trap`** ***`level`***
- **To log to internal buffer use:**
  - **`logging buffered`** ***`size`***
- **Ensure timestamps and sequence numbers**
  - **`service timestamps log…`**
  - **`service sequence-numbers`**

SNMP

Cisco.com

- **Version 1 sends cleartext community strings and has no policy reference**
- **Version 2 addresses some of the known security weaknesses of SNMPv1**
- **Version 3 provides authentication, encryption**
  - **Not widely deployed**
  - **Confirm NMS application support**

SNMP v1/2
Authentication and Authorization

- **Line ACL can filter SNMP access**
- **SNMP Filtering**

  RO → read only

  RW → read write

  View → MIB restriction

```
access-list 4 permit 172.16.2.100
snmp-server community <string> RO 4
snmp-server community <string> view <MIB view>
```

Access to the Router

- **Console**
- **Telnet**
- **SSH—Encrypted Access**
- **Local passwords**
  - **Username based on the router**
  - **Use "enable secret"**
- **External AAA**
  - **TACACS+, RADIUS, Kerberos**
- **One-Time Passwords (OTP)**

Use Enable Secret

- **Service password-encryption is reversible**

```
service password-encryption
!
hostname Router
!
enable password 7 14181C0E2A2B182A2824
```

- **The "enable secret" password hashed via MD5**

```
!
Hostname Router
!
enable secret 5 $1$hM3l$.s/DgJ4TeKdDkTVCJpIBw1
```

VTY Security

- Access to VTYs should be controlled
- ACL used to filter incoming data
- Logging can be used to provide more information

```
access-list 3 permit 192.168.1.0 0.0.0.255
access-list 3 deny   any
line vty 0 4
    access-class 3 in
    transport input ssh
    transport output none
```

SSH

- **Replaces telnet for a protected command and control communication channel**
- **Privacy and integrity provided through the use of strong cryptographic algorithms**
- **Supports TACACS+, RADIUS and Local Authentication**
- **Secure Copy (SCP) available in new SSH enabled code**
- **Restrict access to ssh via transport input ssh command**
- **SSHv2 now in IOS (12.3(4)T / 12.1(19)E)**

Banners

- **Login Banner**

  This is a legal requirement in some jurisdictions; check with your legal group

  ```
  banner login ^
   Authorised access only
   This system is the property of Galactic Internet
   Disconnect IMMEDIATELY if you are not an authorised user!
   Contact noc@isp.net 555-1212 for help.
  ^
  ```

- **Exec Banner**

  Used to remind staff of specific conditions:

  ```
  banner exec ^
  PLEASE NOTE - THIS ROUTER SHOULD NOT HAVE A DEFAULT ROUTE!
  It is used to connect paying peers. These ‘customers’ should not be able to
  default to us.
   The config for this router is NON-STANDARD
   Contact Network Engineering 555-1212 for more info.
  ^
  ```

Cisco IOS TACACS+ Login Authentication

Encrypts Passwords with Encryption (7)

Define List "neteng" to Use TACACS+

Define List "tech" to Use TACACS+ then the Local User and Password

Enable Secret Overrides the (7) Encryption

Define Local Users

Secret Command → md5

```
!
service password-encryption
!
hostname Router
!
aaa new-model
aaa authentication login neteng group tacacs+ enable
aaa authentication login tech group tacacs+ local
aaa authentication enable default group tacacs+ enable
enable secret 5  $1$hM3l$.s/DgJ4TeKdDk…
!
username bill secret 5 $1$A4Um$1NkLTeSwxYynxlHD6zlfc1
```

Cisco IOS TACACS+ Login Authentication


```
tacacs-server host 172.16.1.4
tacacs-server key <key>
!
line con 0
 login authentication neteng
line aux 0
 login authentication neteng
line vty 0 4
 login authentication tech
!
end
```

Defines the IP Address of the TACACS+ Server

Defines the Shared Key for Communicating with the TACACS+ Server

Uses the Authentication Mechanisms Listed in “neteng”—TACACS+ then Enable Password

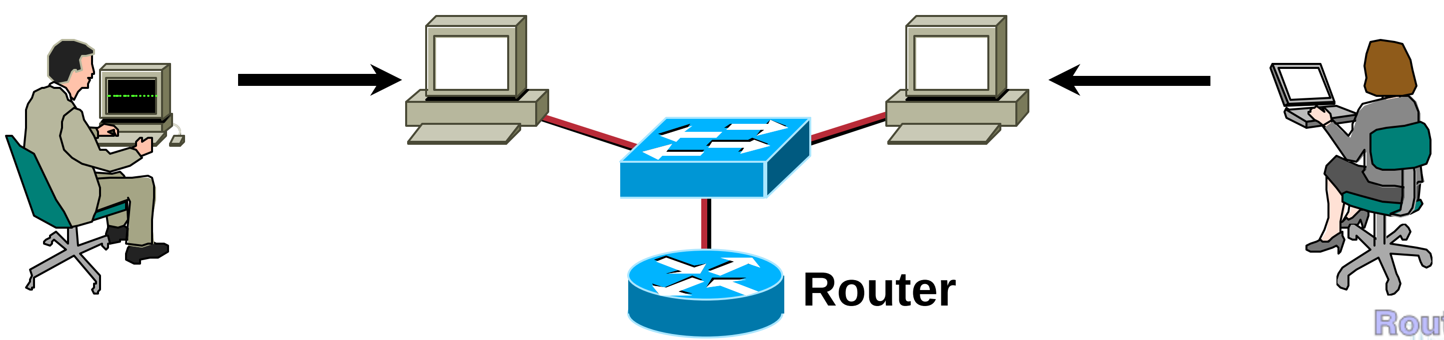
Uses the Authentication Mechanisms Listed in “tech”—TACACS+ then a Local User/Password

Limit Authority—Authorize Commands

- **Differentiate staff authority on the router**
  - **Help desk**
  - **Operations**
  - **Second level/third level support**
- **Use privilege levels (0–15)**

System Administrator
Level 2:
show, debug, ping

Network Engineer
Level 15:
all commands



Set Privileges

Cisco.com

- Set level of privilege for each user class

```
privilege configure level 5 interface
privilege interface level 5 shutdown
privilege exec level 5 show ip route
privilege exec level 5 configure terminal
privilege exec level 5 show running-config
```

- Initially difficult to deploy
- Long-term benefit outweighs short term pain
- Other options are TACACS+-based authorization or…

Role Based CLI Access

- **Role-Based CLI, aka CLI Views**
- **Defines CLI access based on administrative roles**
- **Security**

  Enhances the security of the device by defining the set of CLI commands that are accessible to a particular user

- **Availability**

  Avoids unintentional execution of CLI commands by unauthorized personnel

- **Operational efficiency**

  Prohibits users from viewing CLI commands that are inaccessible to them, greatly improving usability

http://www.cisco.com/en/US/partner/products/sw/iosswrel/ps5207/products_feature_guide09186a00801ee18d.html

Input Hold Queue

- **Queue that stores packets destined for the router**
- **Input Hold Queue is important for initial BGP convergence (when you are sending the full table)**
- **DOS/DDOS attacks against the router can fill the input hold queue—knocking out legitimate packets**

Input Hold Queue

- Input Hold Queue is physically on the Route Processor (RP for 7500, GRP for 12000)
- Default is 75
- Recommend 1500 (Check memory before applying—looking for 20M free) – improves BGP convergence with Internet routing table.
- Applied to all interfaces

```
interface XXXXXX
  hold-queue 1500 in
```

Input Hold Queue


```
12008-e10-2#sh inter pos 5/0
POS5/0 is up, line protocol is up
.
  Output queue 0/40, 0 drops; input queue 97/1500, 54 drops
  5 minute input rate 76502000 bits/sec, 31139 packets/sec
  5 minute output rate 72517000 bits/sec, 26560 packets/sec
.
.
```

Selective Packet Discard (SPD)

- **When a link goes to a saturated state, you will drop packets; the problem is that you will drop any type of packets—including your routing protocols**
- **Selective Packet Discard (SPD) will attempt to drop non-routing packets instead of routing packets when the link is overloaded**

Selective Packet Discard (SPD)

- Input Hold Queue (default 75)
- SPD Headroom (default 100 – in 12.0(22)S increased to 1000)
- SPD Extended Headroom (default 10)

Monitoring SPD Queues

- **You have a problem when you:**
  - **See the number of priority packets drop (H)**
  - **See the Fast Flushes increase (D)**

```
GSR-2#sh interface pos 0/0 switching
POS0/0 Link to GSR#1
          Throttle count         (A)
        Drops             RP     (B)        SP        (C)
   SPD Flushes          Fast     (D)       SSE        (E)
   SPD Aggress          Fast     (F)
  SPD Priority        Inputs     (G)     Drops        (H)
```

Monitoring SPD Modes

- SPD has three drop modes:

  NORMAL—below threshold

  RANDOM—min threshold has been reached

  MAX—max threshold has been reached

- There is a problem when Current Mode is MAX

```
GSR-2#sh ip spd
Current mode: normal.
Queue min/max thresholds: 73/100, Headroom: 1000, Extended Headroom: 100
IP normal queue: 0, priority queue: 0.
SPD special drop mode: aggressively drop bad packets
```

Routing Protocol Security

- Routing protocols can be attacked
  - Denial of service
  - Smoke screens
  - False information
  - Reroute packets
- Protect the routing protocol!
  - Prefix Filtering
  - Routing Protocol Authentication

May Be Accidental or Intentional!

What to Prefix Filter?

Cisco.com

- **Bogons**

  IANA has reserved several blocks of IPv4 that have yet to be allocated to a RIR:

  http://www.iana.org/assignments/ipv4-address-space

- **Special-Use IPv4 Addresses**

  Special Use Addresses (SUA) are reserved for special use :-)

  Defined in RFC3330: ftp://ftp.isi.edu/in-notes/rfc3330.txt

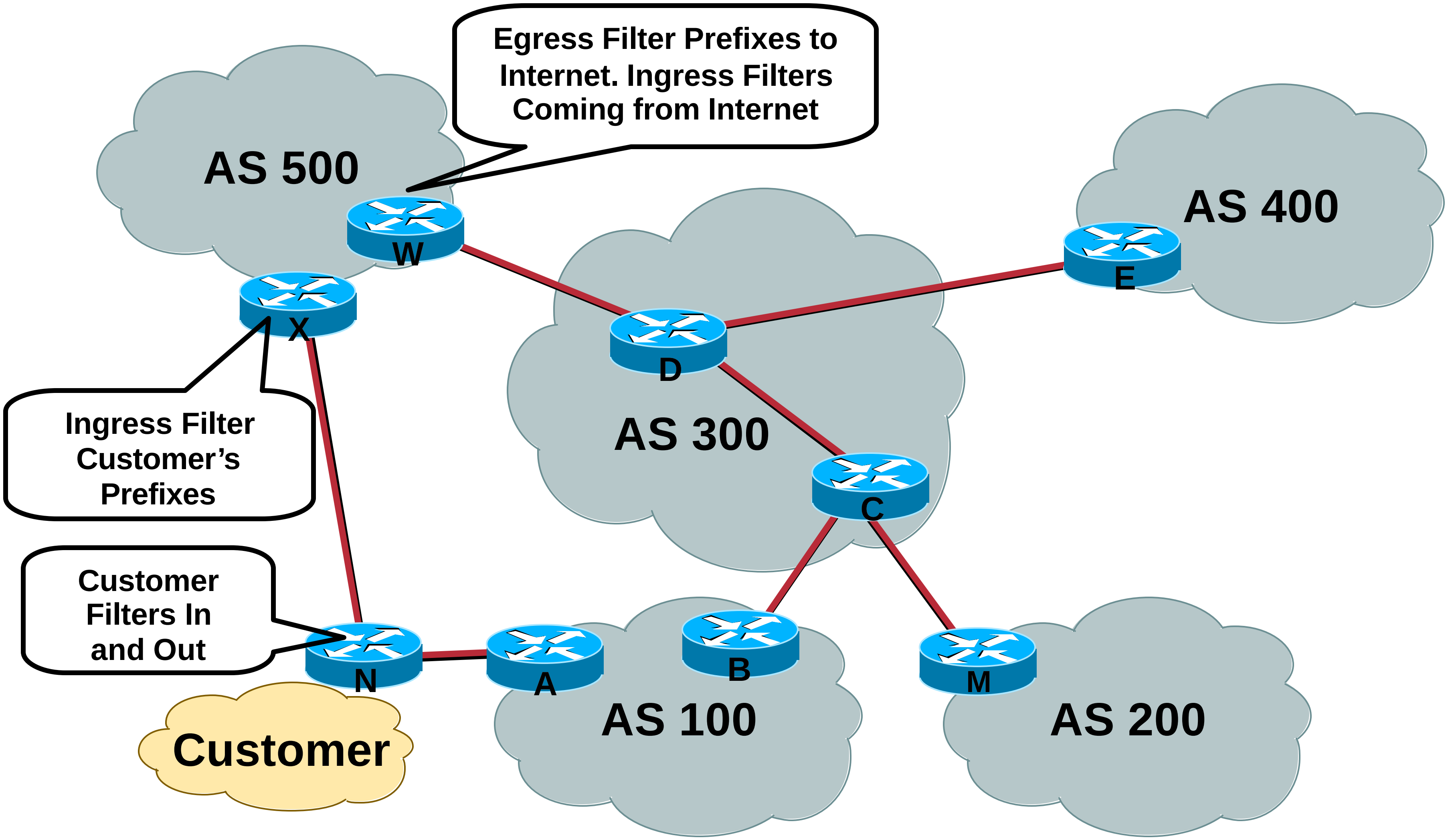
  Examples: 127.0.0.1, 192.0.2.0/24

- **These blocks of IPv4 addresses should never be advertised into the global Internet Route Table**

- **Filters should be applied on the AS border for all inbound and outbound advertisements**

Where to Prefix Filter?

Cisco.com



Routing DNA

New Feature!
BGP Support for TTL Security Check

- **AKA BGP TTL Security Hack (BTSH)**
- **Protects eBGP sessions from CPU attacks using forged IP packets**
- **Not supported for iBGP**
- **Prevents attempts to hijack eBGP session by attacker not part of either BGP network or that is not between the eBGP peers**
- **Minimum Time To Live (TTL) for incoming packets from a specific eBGP peer**
  - **BGP session established and maintained only if TTL in IP packet header is equal to or greater than configured TTL value**
  - **If value is less than configured value packet is silently discarded and no ICMP message generated**
- **Example**

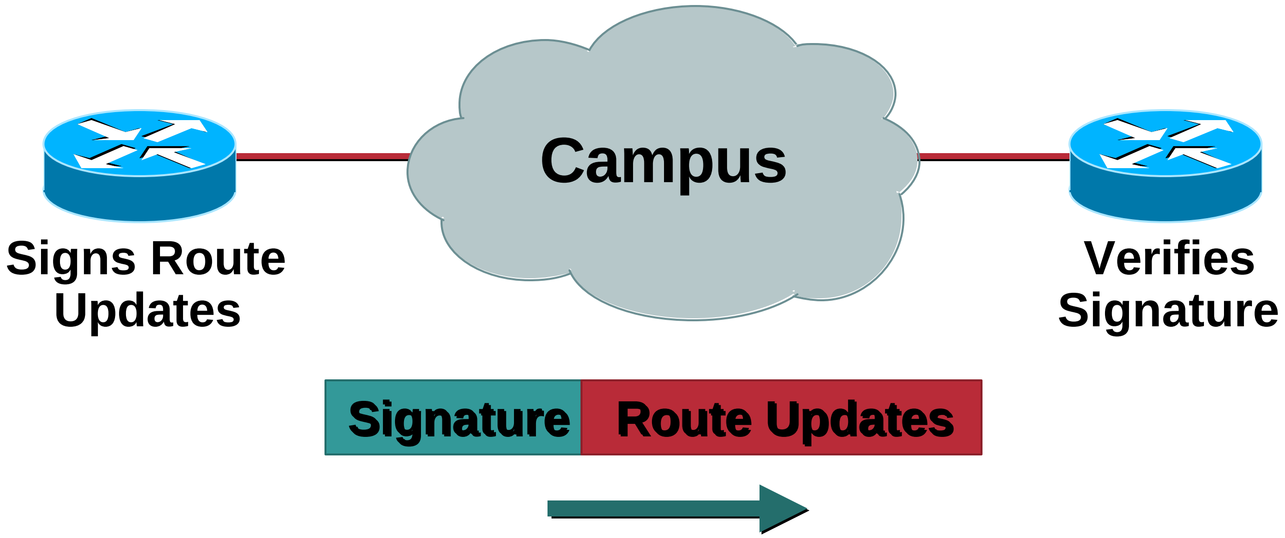
```
router bgp 65530
 neighbor 10.1.1.1 ttl-security hops 2
 ! expected TTL value in the IP packet header is 253
```

- **Available in 12.0(27)S, 12.3(7)T, and 12.2(25)S**

  http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123newft/123t/123t_7/gt_btsh.htm

Secure Routing
Route Authentication

Configure Routing Authentication



Certifies Authenticity of Neighbor and Integrity of Route Updates

Route Authentication

- **Authenticates routing update packets**
- **Shared key included in routing updates**

  Plain text—Protects against accidental problems only

  Message Digest 5 (MD5)—Protects against accidental and intentional problems

- **Often non-implemented**

  "Never seen an attack"

  "My peer doesn't use it"

Route Authentication

- **Multiple keys supported**
  - **Key lifetimes based on time of day**
  - **Use first valid key**
- **Supported for BGP, IS-IS, OSPF, RIPv2, and EIGRP**
- **Syntax differs depending on routing protocol**

OSPF and ISIS Authentication Example

- OSPF

```
interface ethernet1
  ip address 10.1.1.1 255.255.255.0
  ip ospf message-digest-key 100 md5 qa*&gtHH3
!
router ospf 1
  network 10.1.1.0 0.0.0.255 area 0
  area 0 authentication message-digest
```

- ISIS

```
interface ethernet0
  ip address 10.1.1.1 255.255.255.0
  ip router isis
  isis password pe#$rt@s level-2
```

BGP Route Authentication


```
router bgp 200
  no synchronization
  neighbor 10.1.2.1 remote-as 300
  neighbor 10.1.2.1 description Link to Excalabur
  neighbor 10.1.2.1 send-community
  neighbor 10.1.2.1 version 4
  neighbor 10.1.2.1 soft-reconfiguration inbound
  neighbor 10.1.2.1 route-map Community1 out
  neighbor 10.1.2.1 password 7 iuhg9287dhsa7swk
```

BGP Route Authentication

- **Works per neighbor or for an entire peer-group –**
- **Two routers with password mis-match:**

  %TCP-6-BADAUTH: Invalid MD5 digest from [peer's IP address]:11004 to [local router's IP address]:179

- **One router has a password and the other does not:**

  %TCP-6-BADAUTH: No MD5 digest from [peer's IP address]:11003 to [local router's IP address]:179

RFC 2827/BCP 38 Ingress Packet Filtering

Your customers should not be sending any IP packets out to the Internet with a source address other then the address you have allocated to them!

ftp://ftp.isi.edu/in-notes/rfc2827.txt

BCP 38 Packet Filtering Principles

- **Filter as close to the edge as possible**
- **Filter as precisely as possible**
- **Filter both source and destination where possible**

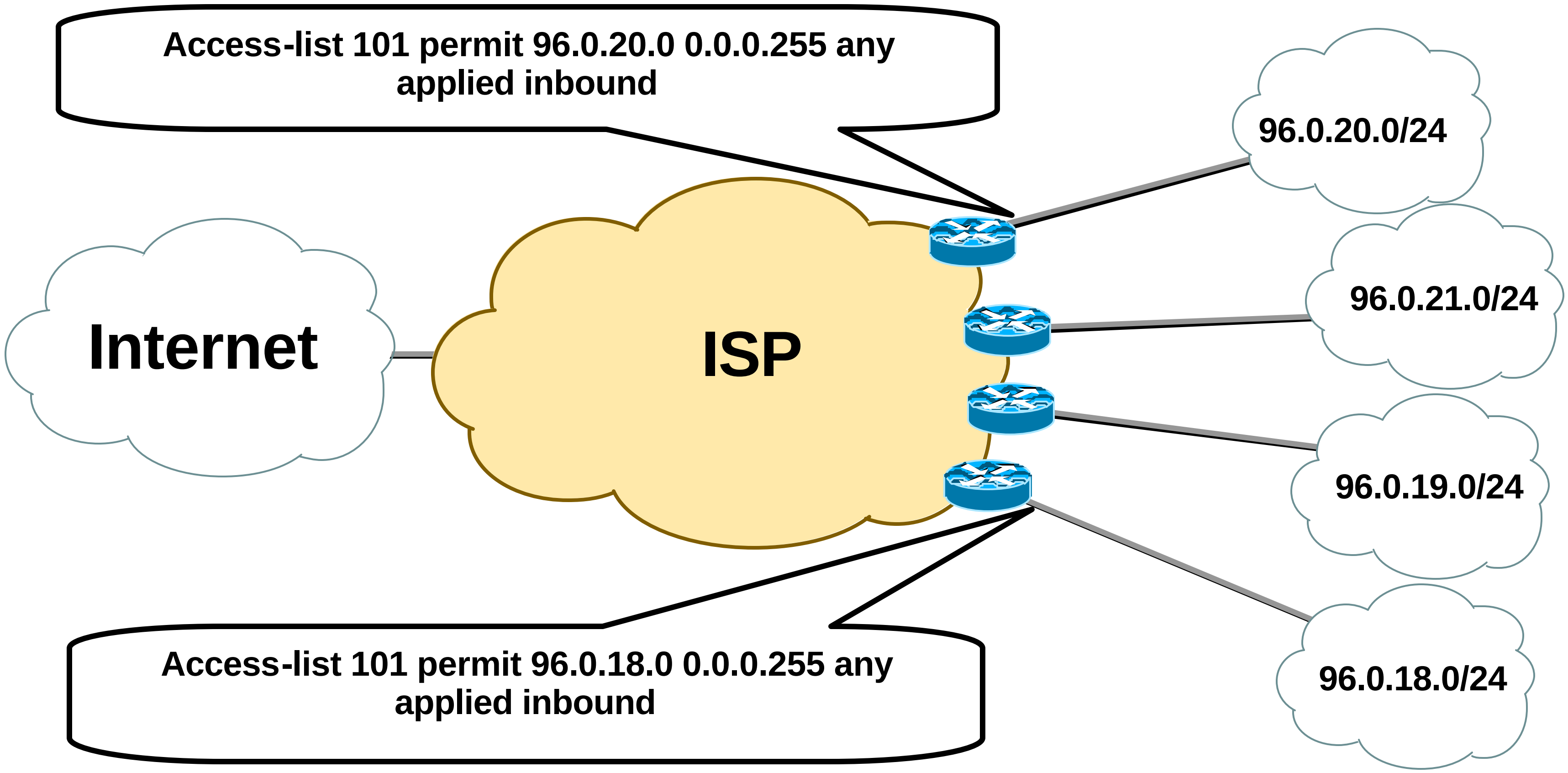
Techniques for BCP 38 Filtering

- **Static ACLs on the edge of the network**
- **Unicast RPF Strict Mode**
- **Cable source verify (DHCP)**
- **Dynamic ACLs with AAA profiles**
- **IP Source Guard**

Static BCP 38 Ingress Packet Filtering

Cisco.com

ISP's Customer Allocation Block: 96.0.0.0/19
BCP 38 Filter = Allow Only Source Addresses from the Customer's 96.0.X.X/24



Unicast Reverse Path Forwarding (uRPF)

- **CEF is required**
- **IP packet source address is checked to ensure that the route back to the source is valid**
- **Two Flavors of uRPF:**

  Strict Mode for:

    BCP 38/RFC 2827 Filters on Customer Ingress Edge

  Loose Mode for:

    ISP-to-ISP Edge

    Remotely Triggered Black Hole Filtering

- **Care required in multihomed situations**

uRPF Strict Mode

A simple and scalable implementation of BCP 38:

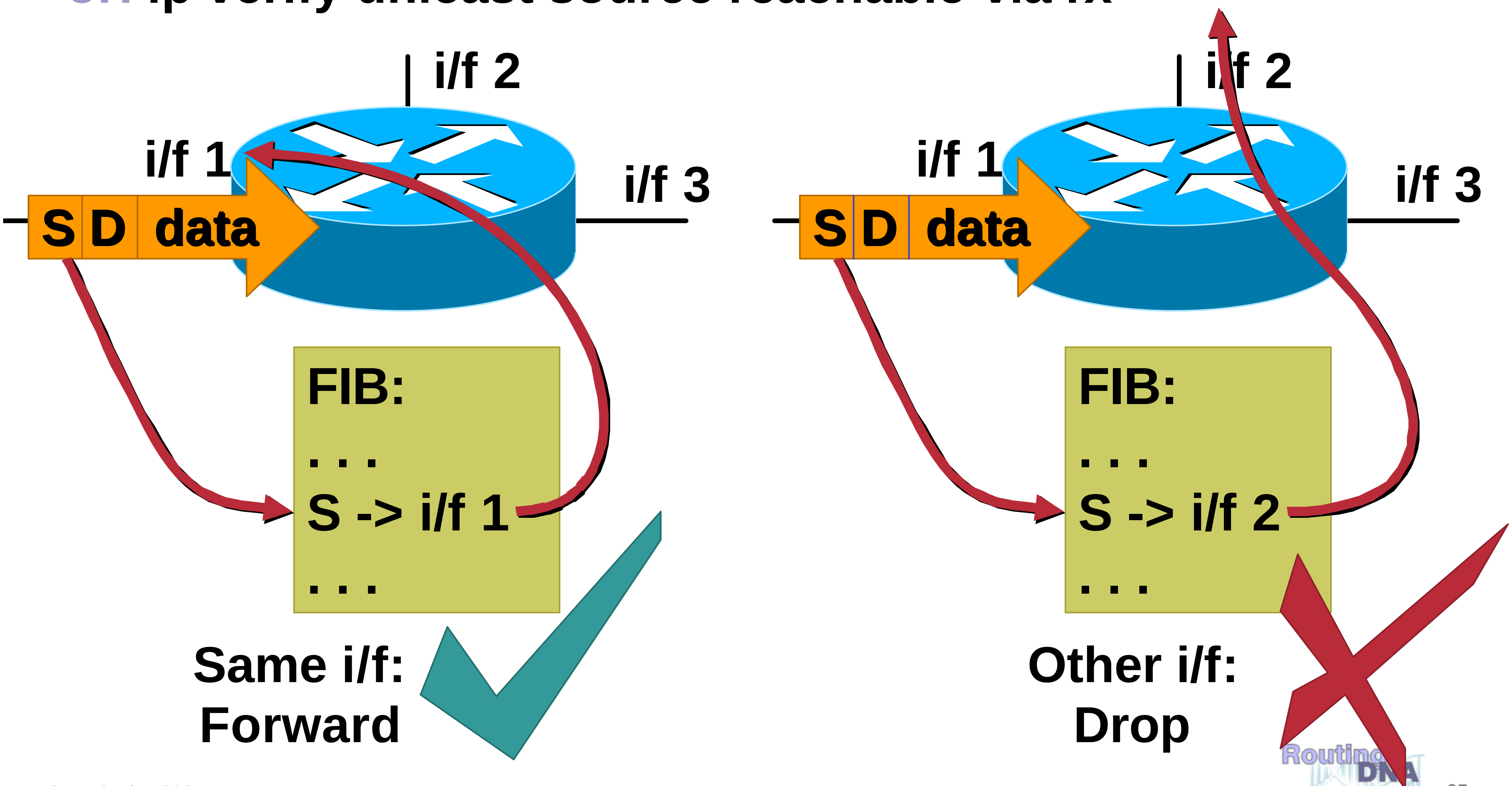
- How do you manage BCP 38 ACLs for over 10,000 lease line customers?
- One command that automatically configures BCP 38 filtering?
- It would be really nice if the line engineer who first brings up the customer interface can configure this feature without needing to create ACLs or touch the routing protocols!
- It would be nice if the filter could be automatically updated!

→ Use uRPF!!!

Strict uRPF Check
(Unicast Reverse Path Forwarding)

Cisco.com

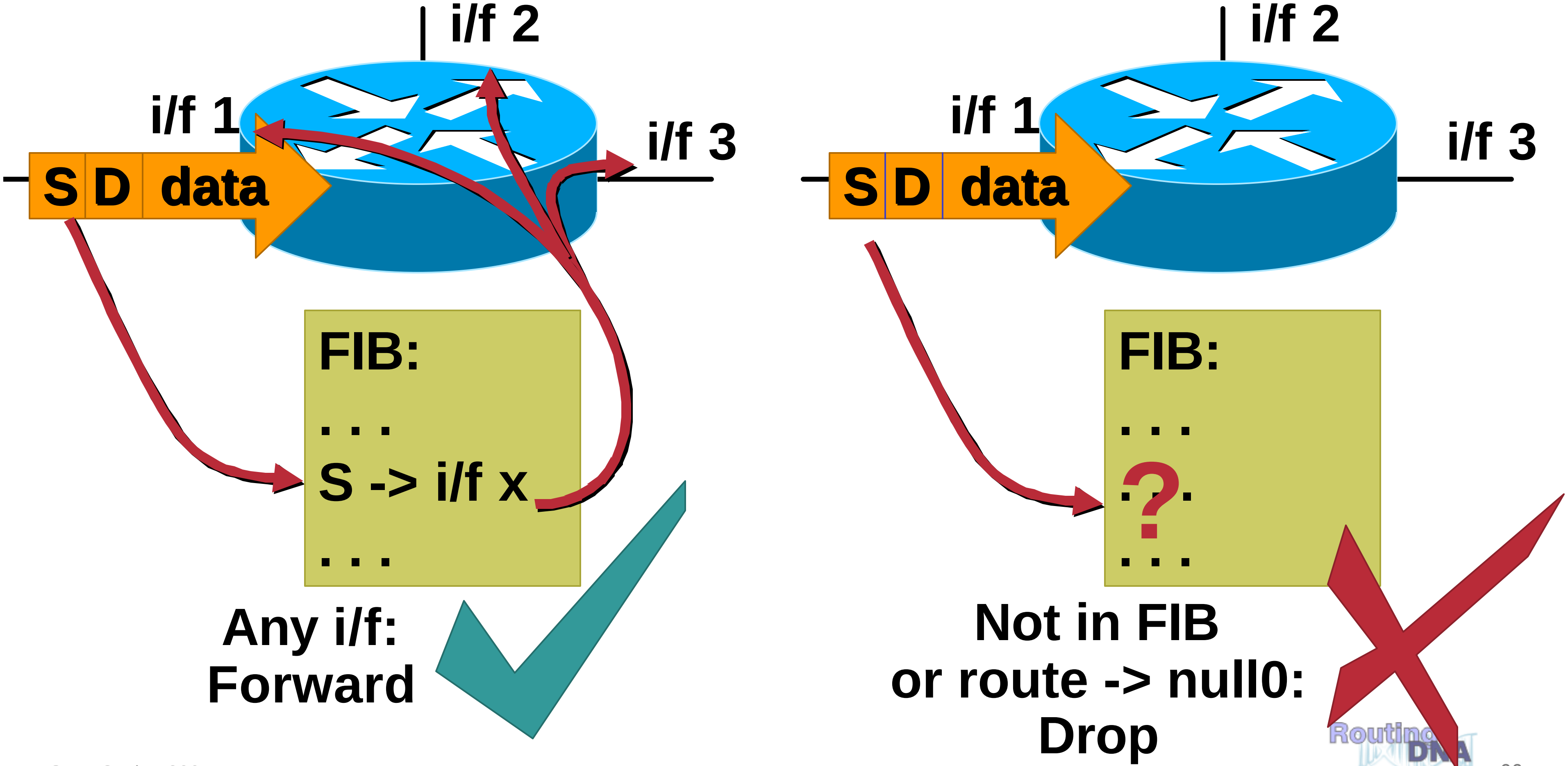
router(config-if)# **ip verify unicast reverse-path**
or: **ip verify unicast source reachable-via rx**



Loose uRPF Check
(Unicast Reverse Path Forwarding)

Cisco.com

router(config-if)# ip verify unicast source reachable-via any



Deploying uRPF

- **Single-homed Customers**
  - uRPF provides simple, easy way to deploy BCP 38 filtering
  - Simple config for many customers
- **Dual-homed Customers**
  - Asymmetric Routing → Must “tweak” routing
  - Use BGP Weight, local_pref to ensure consistent best path
  - uRPF can be used with dual homed customers with proper engineering

Unicast RPF Verification

Cisco.com

Commands:

show ip traffic | include RPF

show ip interface ethernet 0/1/1 | include RPF

debug ip cef drops rpf <ACL>

```
Router# show ip traffic
   IP statistics:
     Rcvd:  1471590 total, 887368 local destination
            …
     Drop:  3 encapsulation failed, 0 unresolved, 0 no adjacency
            0 no route, 0 unicast RPF, 0 forced drop
```

Agenda

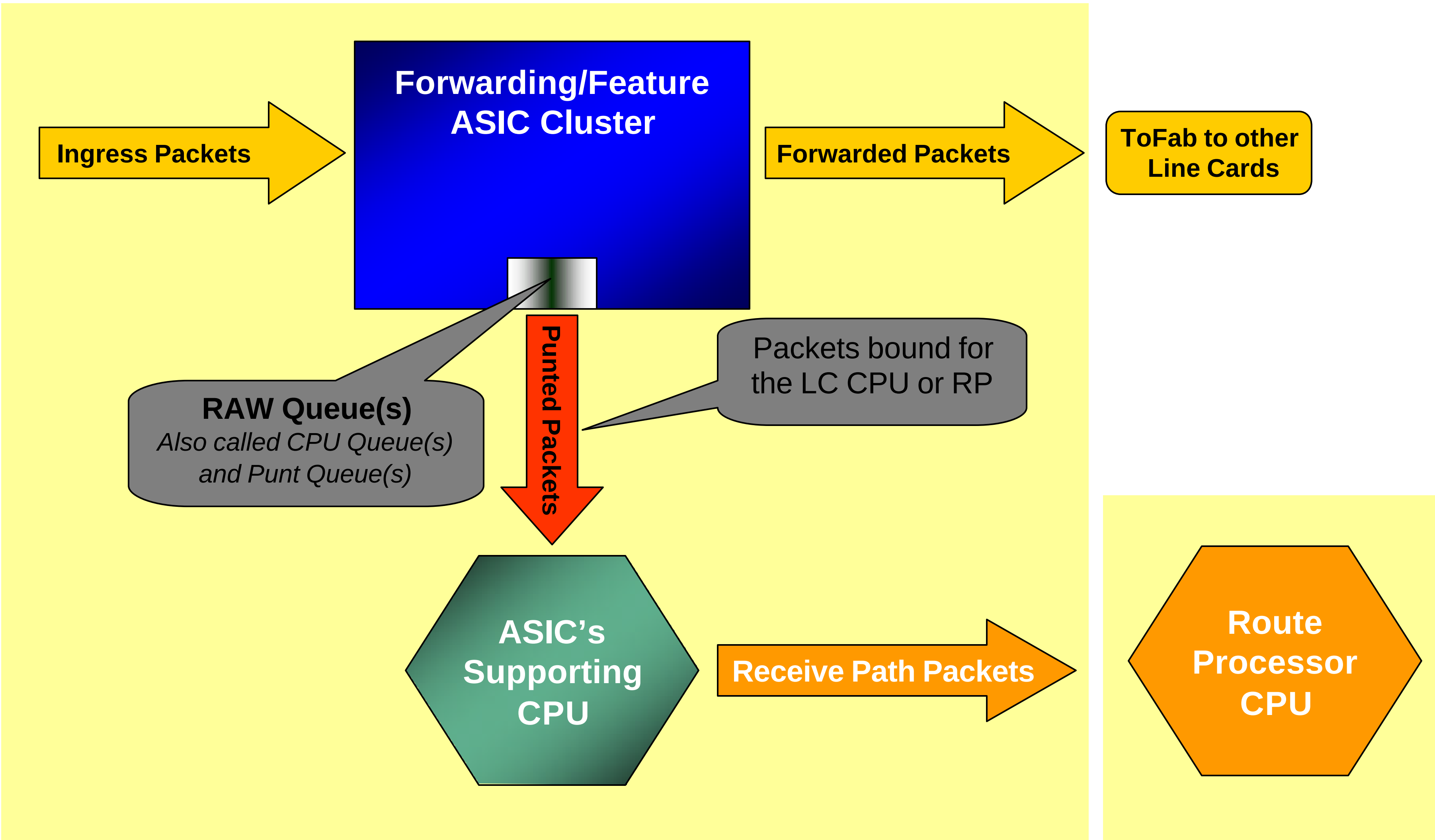
Cisco.com

- Infrastructure security overview
- Preparing The Network
- Router Security: A Plane Perspective
- Tools and Techniques
- Platform Architecture
- Conclusions

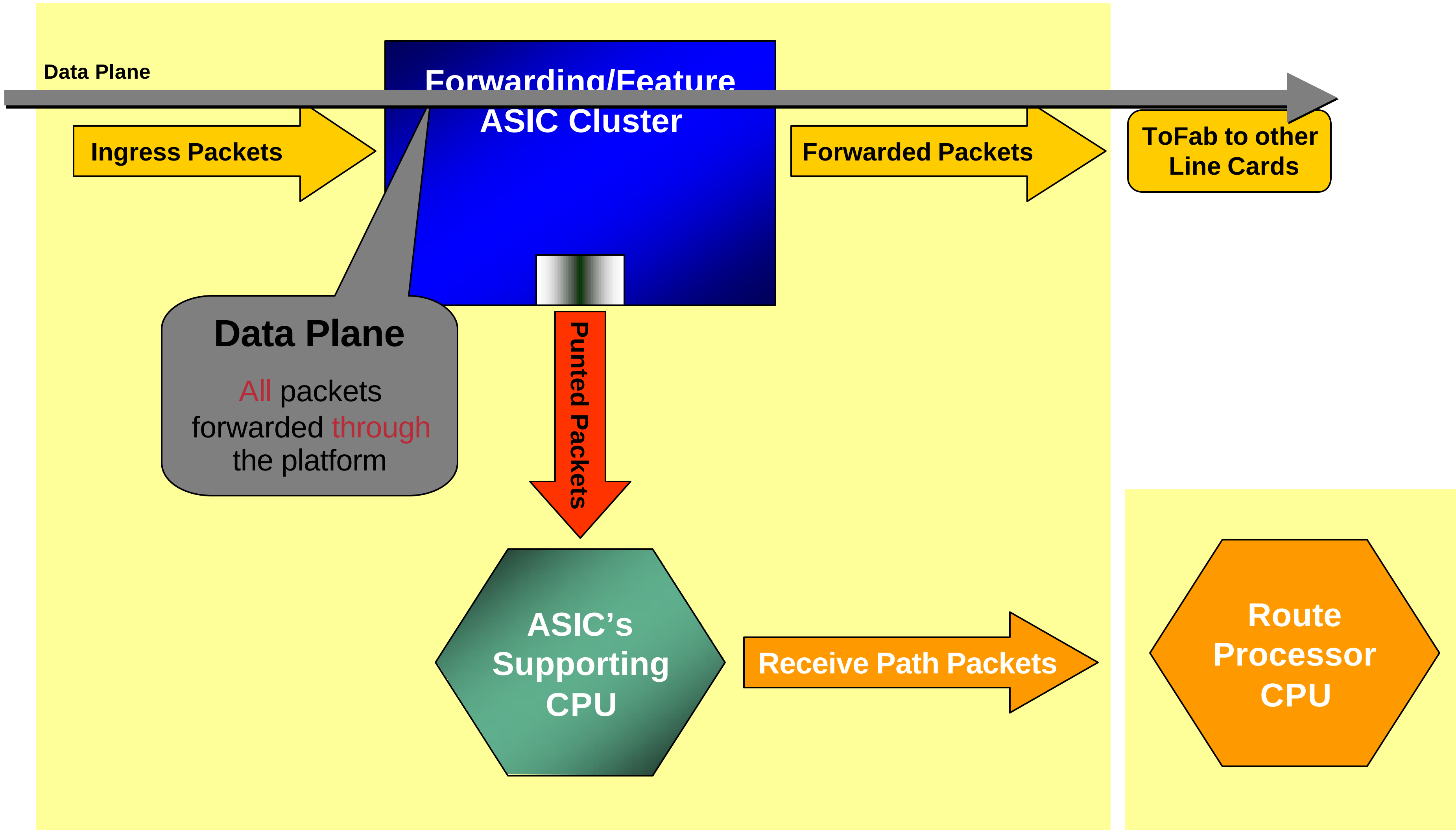
Routers and Planes

- A network device typically handles traffic in the data/forwarding plane, the control plane, and the management Plane
- Traffic in the data/forwarding plane is always destined through the device, and is:
  - Implemented in hardware on high end platforms
  - CEF switched (in the interrupt) in software switched platforms
- Traffic to the control/management plane is always destined to the device and is handled at process level ultimately:
  - In hardware switched platforms, control/management plane traffic is sent to the RP/MFSC and then sent to the process level for processing
  - In software switched platforms, it is sent directly to the process level for processing
- Some data plane traffic also reaches the control plane
  - Packets that are not routable reach to control plane so that ICMP unreachable messages can be generated
  - Packets that have IP options set are also handled by the processor

ASIC Based Platform – Main Components

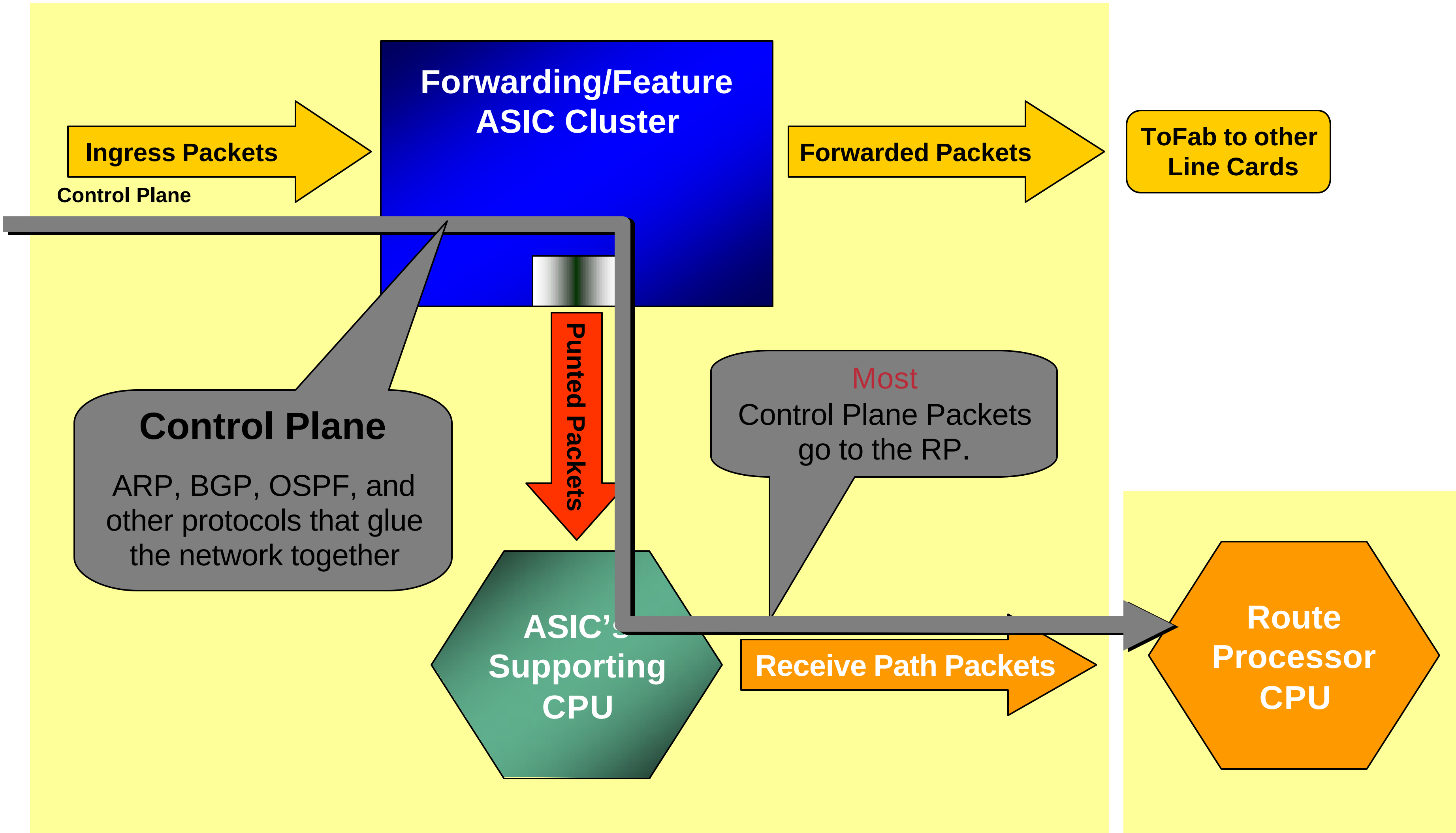


Data Plane

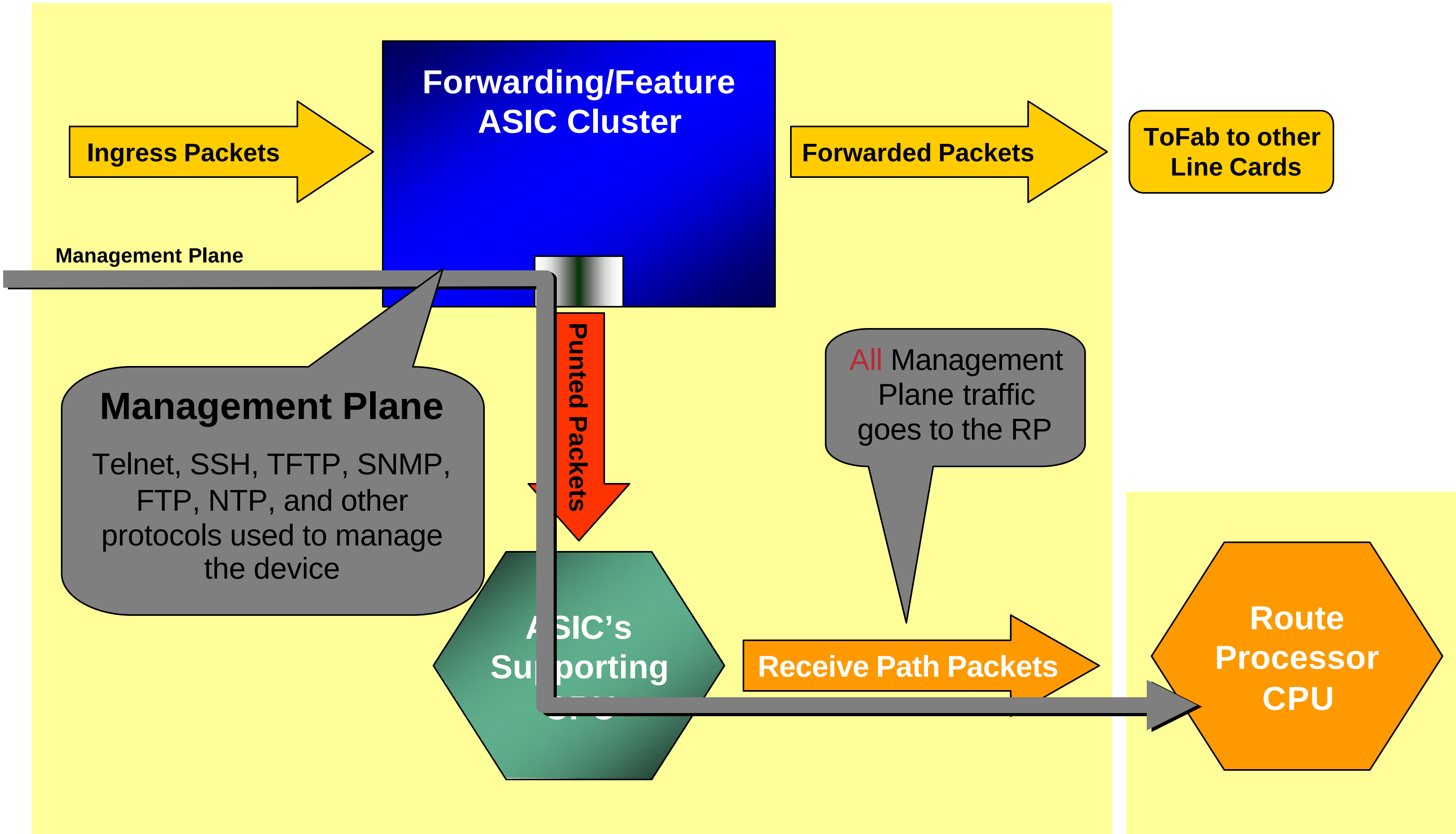


Data Plane
Forwarding/Feature ASIC Cluster
Ingress Packets
Forwarded Packets
ToFab to other Line Cards
Data Plane
All packets forwarded through the platform
Punted Packets
ASIC's Supporting CPU
Receive Path Packets
Route Processor CPU

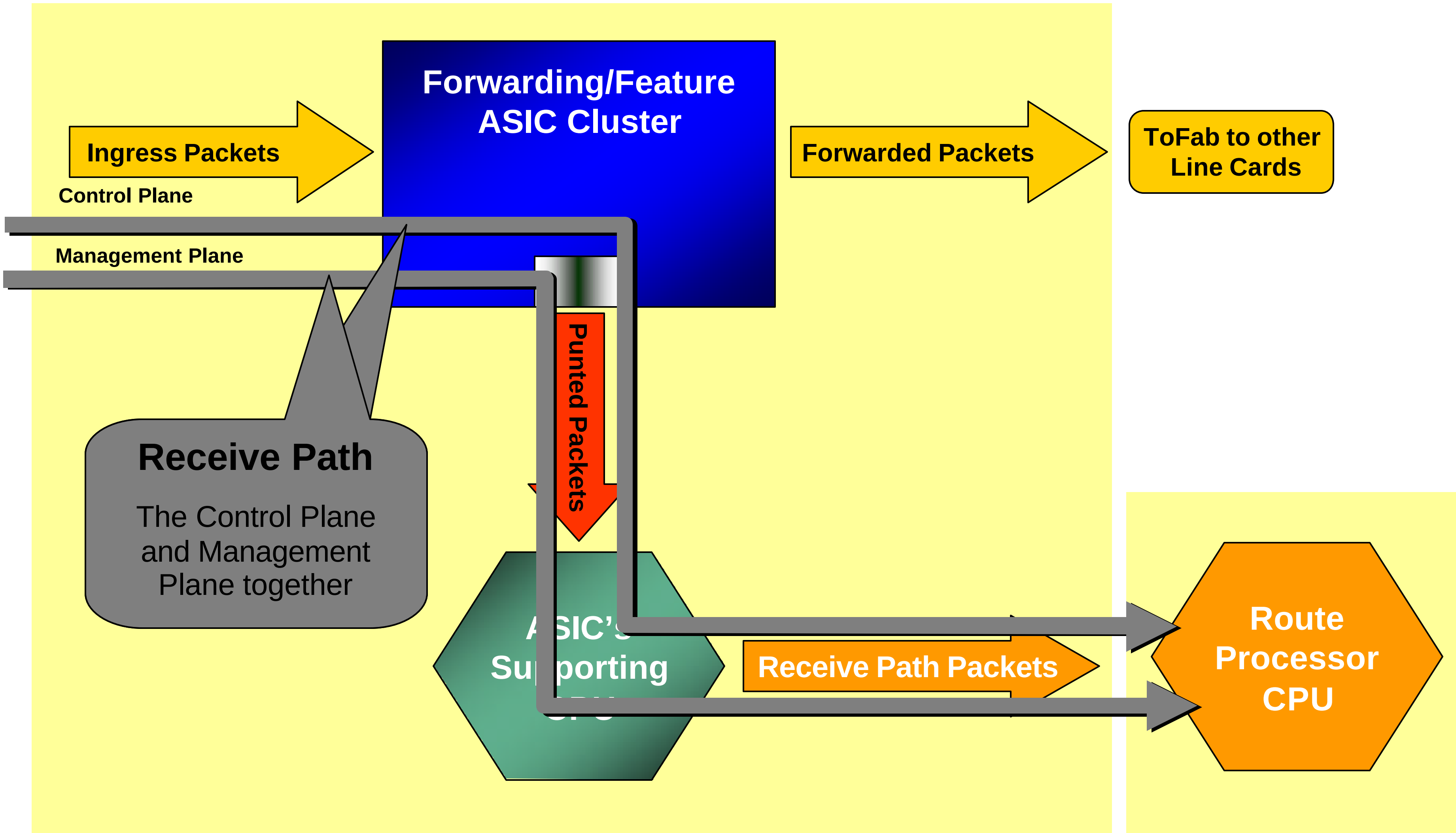
Control Plane



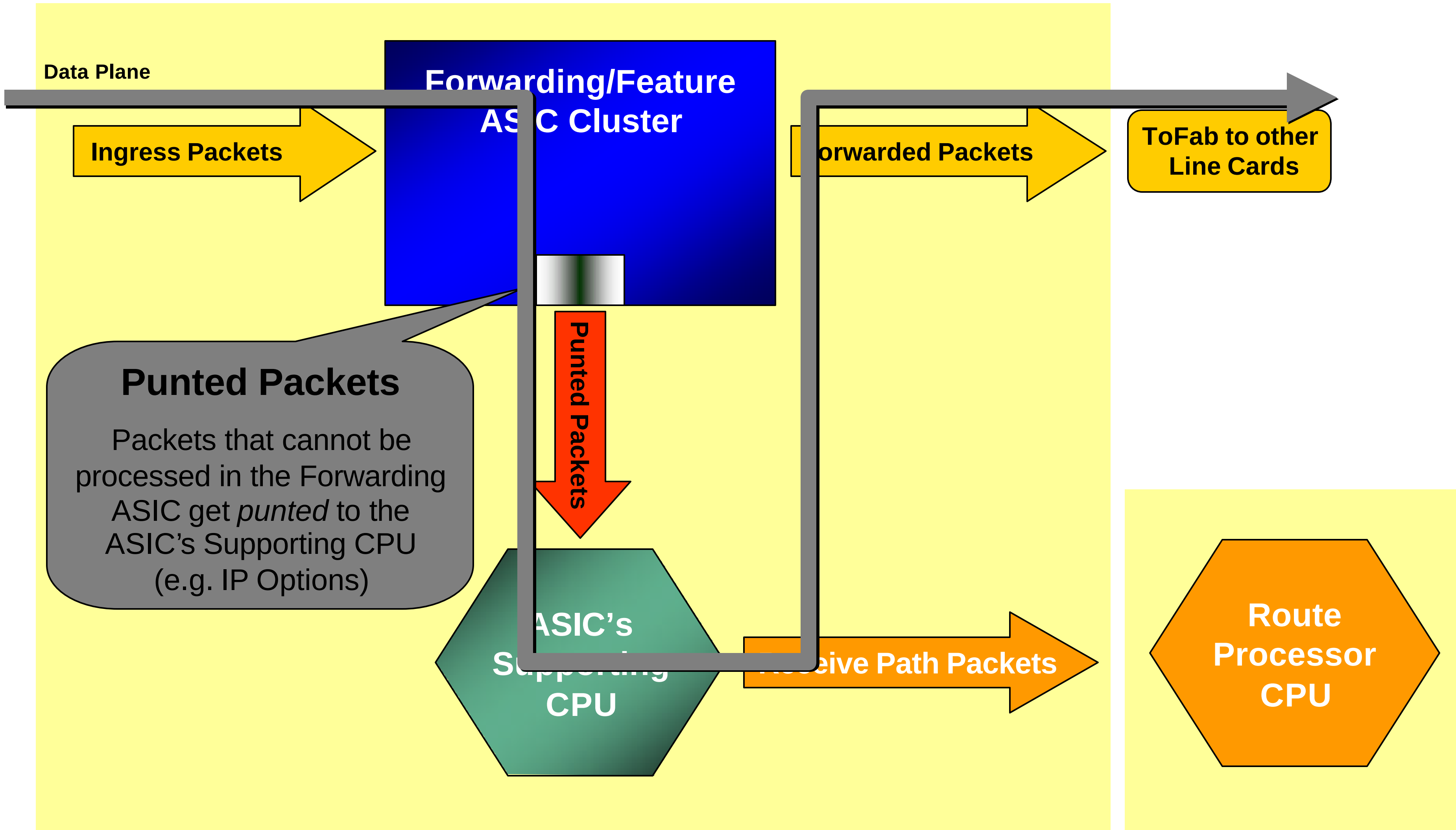
Management Plane



Receive Path

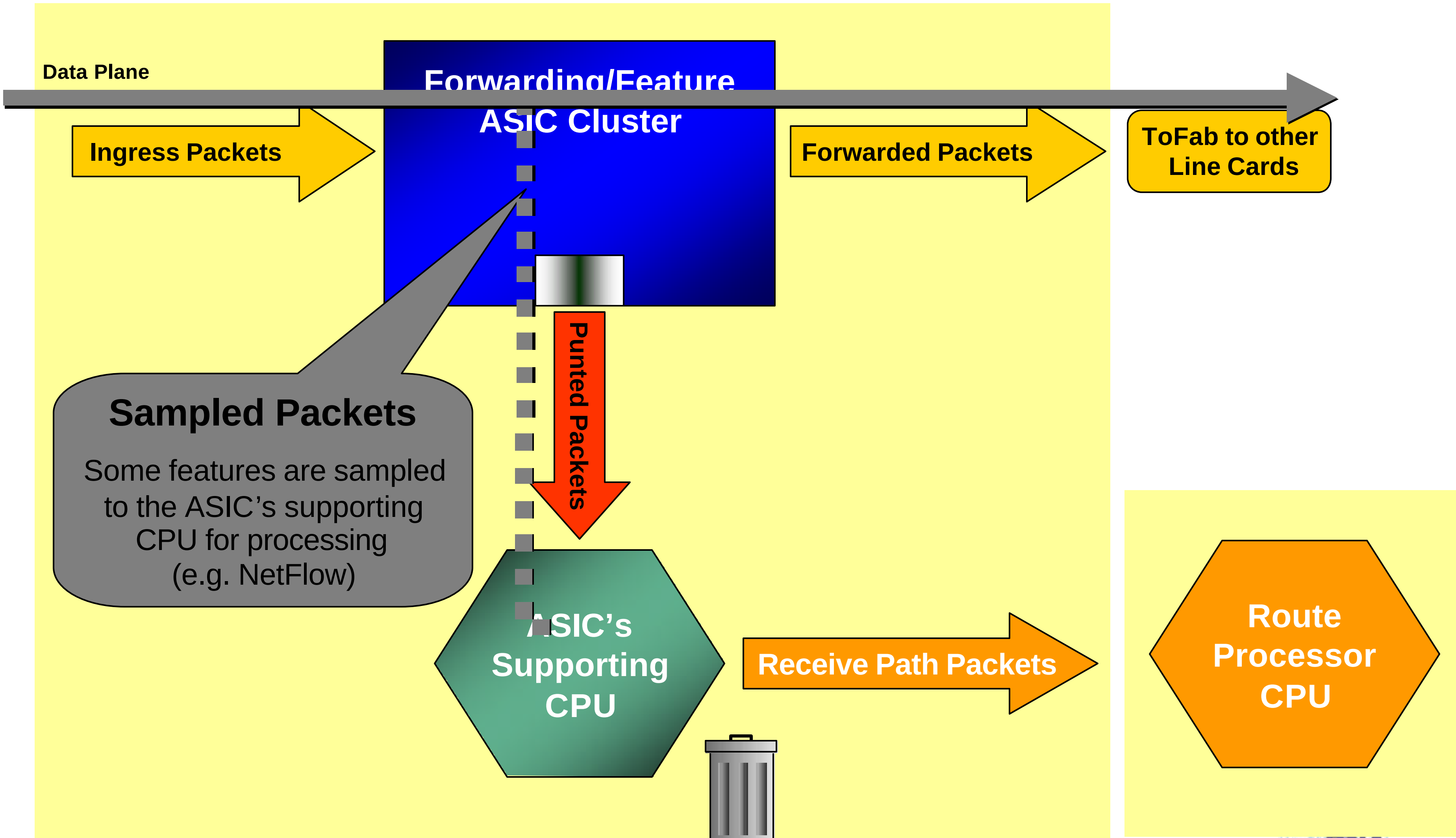


Feature Punt



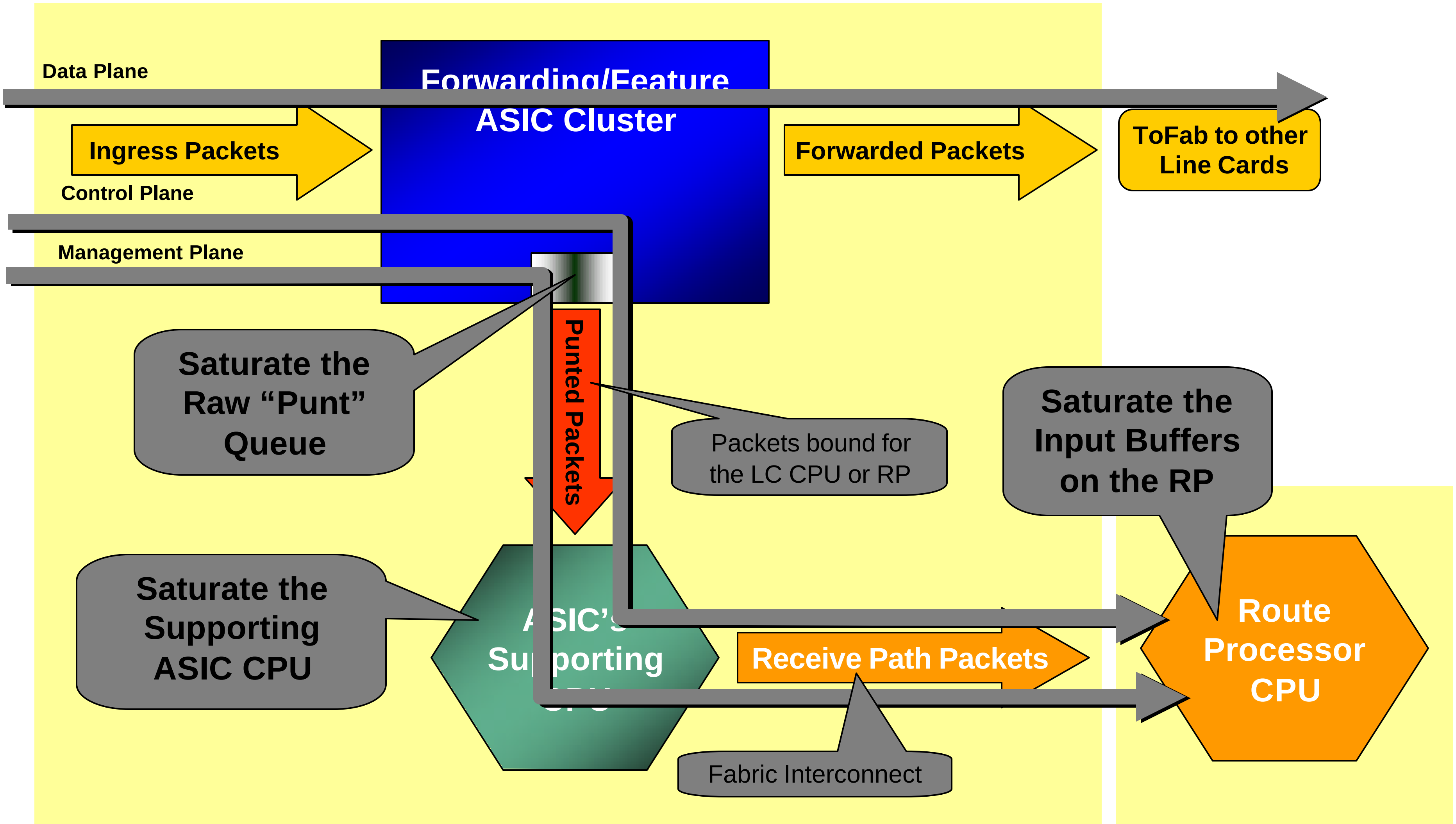
Sampled Feature

Cisco.com



Receive Path Attack Vectors

Cisco.com



Router Risk Assessment

- **Direct router attacks usually target:**
  - Bandwidth saturation (data plane)
  - Control and/or management plane (receive path traffic on the control and management plane)
  - Saturate the punt path out of the forwarding/feature ASIC by abusing the TCP/IP standards (data plane traffic that is punted from the forwarding/feature ASIC).
- **High level of Control Plane activity can cause various side effects**
  - High route processor CPU utilization (near 100%)
  - Loss of keep-alives & routing protocol updates
  - Route flaps and major network transitions
  - Indiscriminate packet drops of incoming packets when memory and buffers are unavailable for legitimate IP data packets
  - Slow or unresponsive interactive sessions via Command Line Interface (CLI)
- **Attacks can be intentional or unintentional**

Agenda

- Infrastructure security overview
- Preparing The Network
- Router Security: A Plane Perspective
- Tools and Techniques
- Platform Architecture
- Conclusions

Taking a Measured Approach

- The techniques we will be discussing are extremely useful, but they must be applied in an architecturally-sound, situationally-appropriate, and operationally-feasible manner
- Don't try to do all this at once—pick a technique with which you are comfortable and which you think will benefit you the most, and start there
- Pilot your chosen technique in a controlled manner, in a designated portion of your network
- Take the lessons learned from the pilot and work them into your general deployment plan and operational guidelines
- Rinse, repeat!

Control Plane Protection Evolution

- **Infrastructure ACLs (iACLs)**
  - **Create policies (ACLs or MQC) for control plane traffic to block all unwanted IP traffic destined to the core**
  - **Applied to ALL ingress port - affects ALL traffic (control and data plane)**
- **Receive Path ACLs (rACLs)**
  - **Create ACLs to block all all unwanted IP traffic destined to the core**
  - **Global (single) configuration affects all “receive path” packets**
  - **Only affects control plane traffic**
- **Control Plane Policing (CoPP)**
  - **Extends rACLs by adding Modular QoS CLI (MQC) policing**
  - **Widespread platform support**

INFRASTRUCTURE ACLs

Infrastructure ACLs

- **Basic premise: filter traffic destined TO your core routers**
  - **Do your core routers really need to process all kinds of garbage?**
- **Develop list of required protocols that are sourced from outside your AS and access core routers**
  - **Example: eBGP peering, GRE, IPSec, etc.**
  - **Use classification ACL as required**
- **Identify core address block(s)**
  - **This is the protected address space**
  - **Summarization is critical → simpler and shorter ACLs**

Infrastructure ACLs

- Infrastructure ACL will permit only required protocols and deny ALL others to infrastructure space
- ACL should also provide anti-spoof filtering
  - Deny your space from external sources
  - Deny RFC1918 space
  - Deny multicast sources addresses (224/4)
  - RFC3330 defines special use IPv4 addressing

A Digression: IP Fragments and Security

- **Fragmented Packets can cause problems…**
  - Fragmented packets can be used as an attack vector by attempting to bypass ACL's
  - Fragments can increase the effectiveness of some attacks by making the recipient consume more resources (CPU and memory) due to fragmentation reassembly
- **ACL fragment handling…**
  - By default (without the *fragments* keyword)…
    - Initial fragments and non-fragmented packets
      - L3 ACL's - ACL action executed (permit/deny) since all L3 information is available
      - L4 ACL's - ACL action executed (permit/deny) since all L4 information is available
    - Non-initial fragment packets (assuming L3 match)
      - L3 ACL's - ACL action executed (permit/deny) since all L3 information is available
      - L4 ACL's - ACL action executed (permit/deny) since all L3 information is available, if the IP header "next layer up" protocol matches the ACL L4 protocol (e.g. IP layer says "6" and ACL is for TCP…)
  - The ACL *fragments* keyword enables specialized handling behavior…
    - Initial fragments and non-fragmented packets
      - L3 and L4 ACL's - assuming an L3 match, if the action is "permit" or "deny," the ACL is - ignored (it's not a match actually, it doesn't match "fragments" keyword) and the next ACL entry is checked…
    - Non-initial fragment packets (assuming L3 match)
      - with L3 and L4 ACL's - assuming an L3 match (and "next layer up" protocol matches the L4 protocol), the action of the ACL is executed (permit/deny)…

iACLs and Fragments

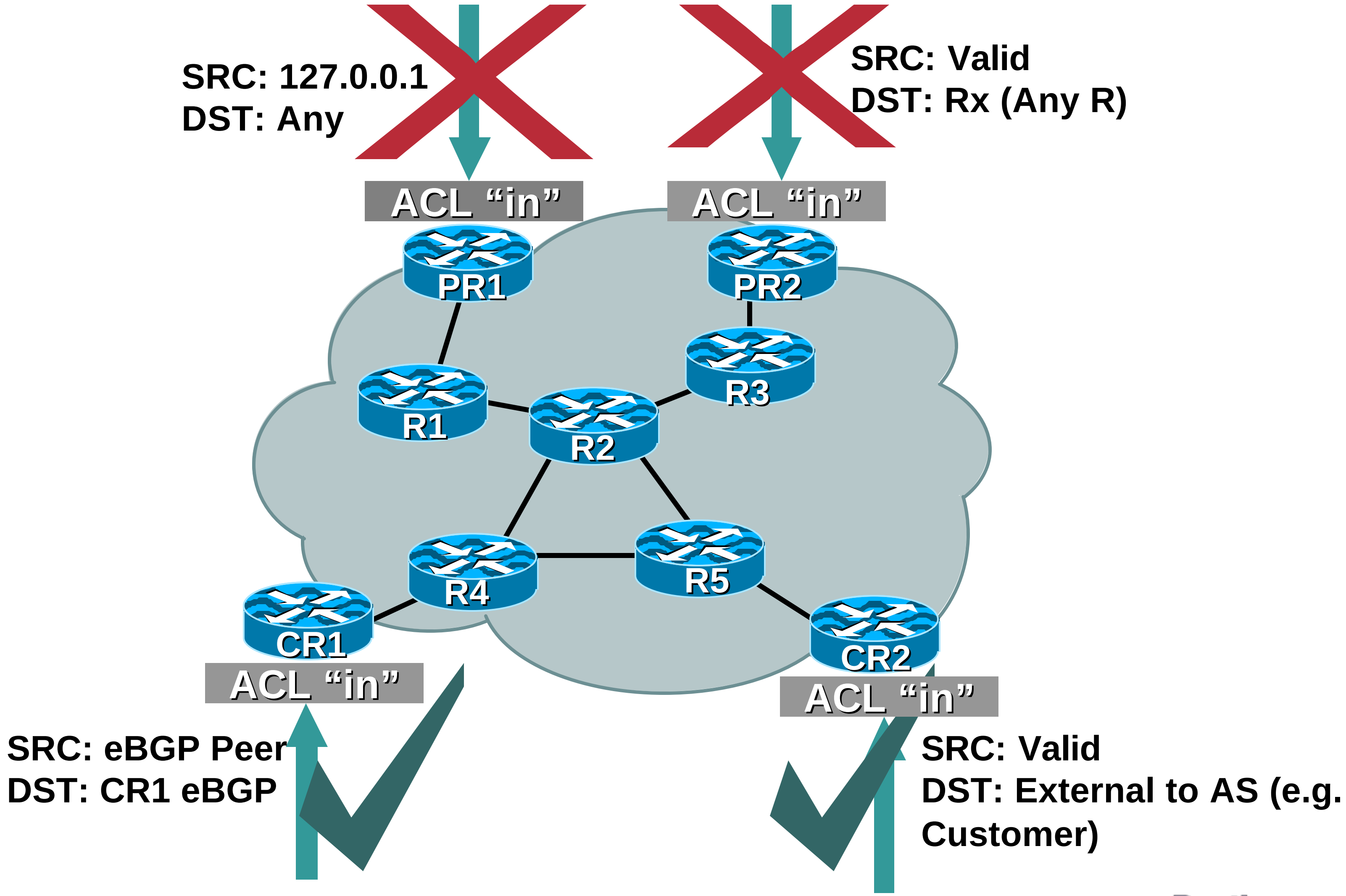
- **Fragments can be denied via an iACL**
- **Denies fragments and classifies fragment by protocol:**

```
access-list 110 deny tcp any core_CIDR fragments
access-list 110 deny udp any core_CIDR fragments
access-list 110 deny icmp any core_CIDR fragments
```

Infrastructure ACLs

- **Infrastructure ACL must permit transit traffic**
  - **Traffic passing through routers must be allowed via permit IP any any**
- **ACL is applied inbound on ingress interfaces**
- **Fragments destined to the core can be filtered via fragments keyword**

Infrastructure ACL in Action



IP Options

Cisco.com

- **Provide control functions that may be required in some situations but unnecessary for most common IP communications**
- **IP Options not switched in hardware**
- **Complete list and description of IP Options in RFC 791**
- **Drop and ignore reduce load on the route processor (RP)**
- **Caution: some protocols/application require options to function:**

  For example: strict/loose source routing, resource reservation protocols (RSVP) and others

- **ip access-list extended drop-ip-option**

  deny ip any any option any-options

  permit ip any any

- **ip options drop**
- **ip options ignore – router ignores options**
  - **Best practice when router doesn't need to process options**
  - **"ignore" not available on all routing platforms**

  Available in 12.0(22)S, 12.3(4)T and 12.2(25)S

  http://www.cisco.com/en/US/products/sw/iosswrel/ps1829/products_feature_guide09186a00801d4a94.html

Other iACL Possibilities

- Edge QoS enforcement

  Control what traffic is "important" in your network

  Don't let attackers take advantage of QoS – re-color at network ingress.

  Philosophical debate for some

  6/7 is easy!

- Rate limiting

  What about letting some traffic in but at a limited rate?

Iterative Deployment

- Typically a very limited subset of protocols needs access to infrastructure equipment
- Even fewer are sourced from outside your AS
- Identify required protocols via classification ACL
- Deploy and test your ACLs

Step 1: Classification

- **Traffic destined to the core must be classified**
- **NetFlow can be used to classify traffic**

  Need to export and review
- **Classification ACL can be used to identify required protocols**

  Series of permit statements that provide insight into required protocols

  Initially, many protocols can be permitted, only required ones permitted in next step

  Log keyword can be used for additional detail. Hits to ACL entry with *log* will increase CPU utilization. Impact varies by platform.

  Consider:

  ```
  Router(config)# ip access-list logging interval <interval ms>
  ```
- **Regardless of method, unexpected results should be carefully analyzed → do not permit protocols that you can't explain!**

Step 2: Begin to Filter

- Permit protocols identified in step 1 to infrastructure only address blocks
- Deny all other to addresses blocks
  - Watch access control entry (ACE) counters
  - Log keyword can help identify protocols that have been denied but are needed
- Last line: permit ip any any ← permit transit traffic
- The ACL now provides basic protection and can be used to ensure that the correct suite of protocols has been permitted

Steps 3 and 4: Restrict Source Addresses

- **Step 3:**
  - ACL is providing basic protection
  - Required protocols permitted, all other denied
  - Identify source addresses and permit only those sources for requires protocols
  - e.g. external BGP peers, tunnel end points
- **Step 4:**
  - Increase security: deploy destination address filters if possible

Example: Infrastructure ACL

! Deny our internal space as a source of external packets

```
access-list 101 deny ip our_CIDR_block any
```

! Deny src addresses of 0.0.0.0 and 127/8

```
access-list 101 deny ip host 0.0.0.0 any
access-list 101 deny ip 127.0.0.0 0.255.255.255 any
```

! Deny RFC1918 space from entering AS

```
access-list 101 deny ip 10.0.0.0 0.255.255.255 any
access-list 101 deny ip 172.16.0.0 0.0.15.255 any
access-list 101 deny ip 192.168.0.0 0.0.255.255 any
```

Example: Infrastructure ACL

! The only protocol that require infrastructure access is eBGP. WE have defined both src and dst addresses

```
access-list 101 permit tcp host peerA host peerB eq 179
access-list 101 permit tcp host peerA eq 179 host peerB
```

! Deny all other access to infrastructure

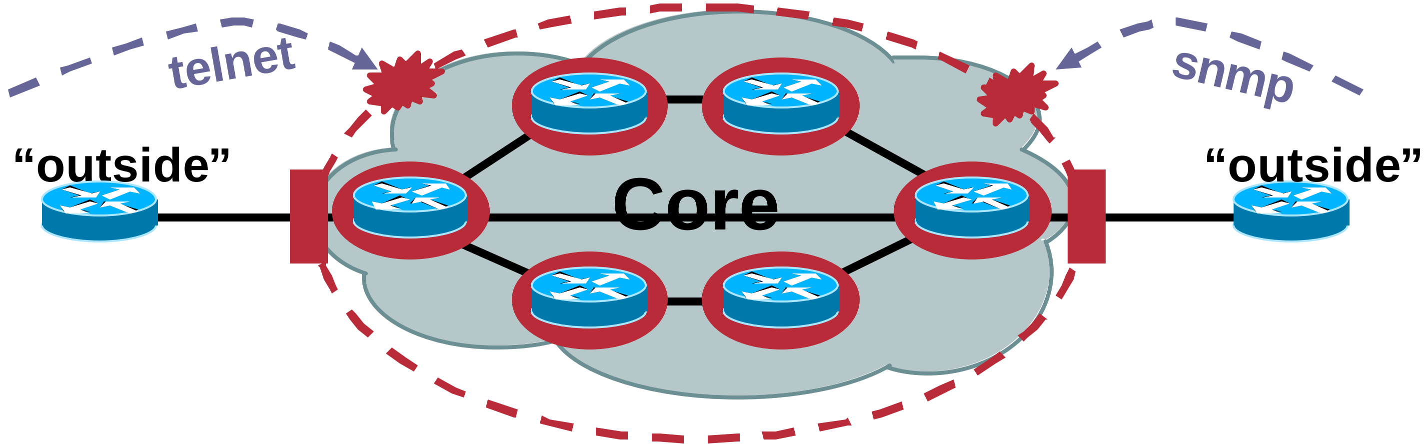
access-list 101 deny ip any core_CIDR_block

! Permit all data plane traffic

```
access-list 101 permit ip any any
```

Infrastructure ACLs

Cisco.com



- **Edge "shield" in place**
- **Not perfect, but a very effective first round of defense**

  Can you apply iACLs everywhere?

  What about packets that you cannot filter with iACLs?

  Hardware limitations

- **Next step: secure the control/management planes per box**

Receive Access-Control List (rACL)

Receive ACLs (rACLs)

- Receive ACLs filter traffic destined to the RP via receive adjacencies
- rACLs explicitly permit or deny traffic destined to the RP
- rACLs do NOT affect transit traffic
- Traffic is filtered on the ingress line card (LC), prior to route processor (RP) processing
- rACLs enforce security policy by filtering who/what can access the router

Receive ACL Command

- **Introduced in 12.0(21)S2/12.0(22)S**

  ip receive access-list [number]

- **Standard, extended or compiled ACL**
- **As with other ACL types, show access-list provide ACE hit counts**
- **Log keyword can be used for more detail**

Receive Adjacencies

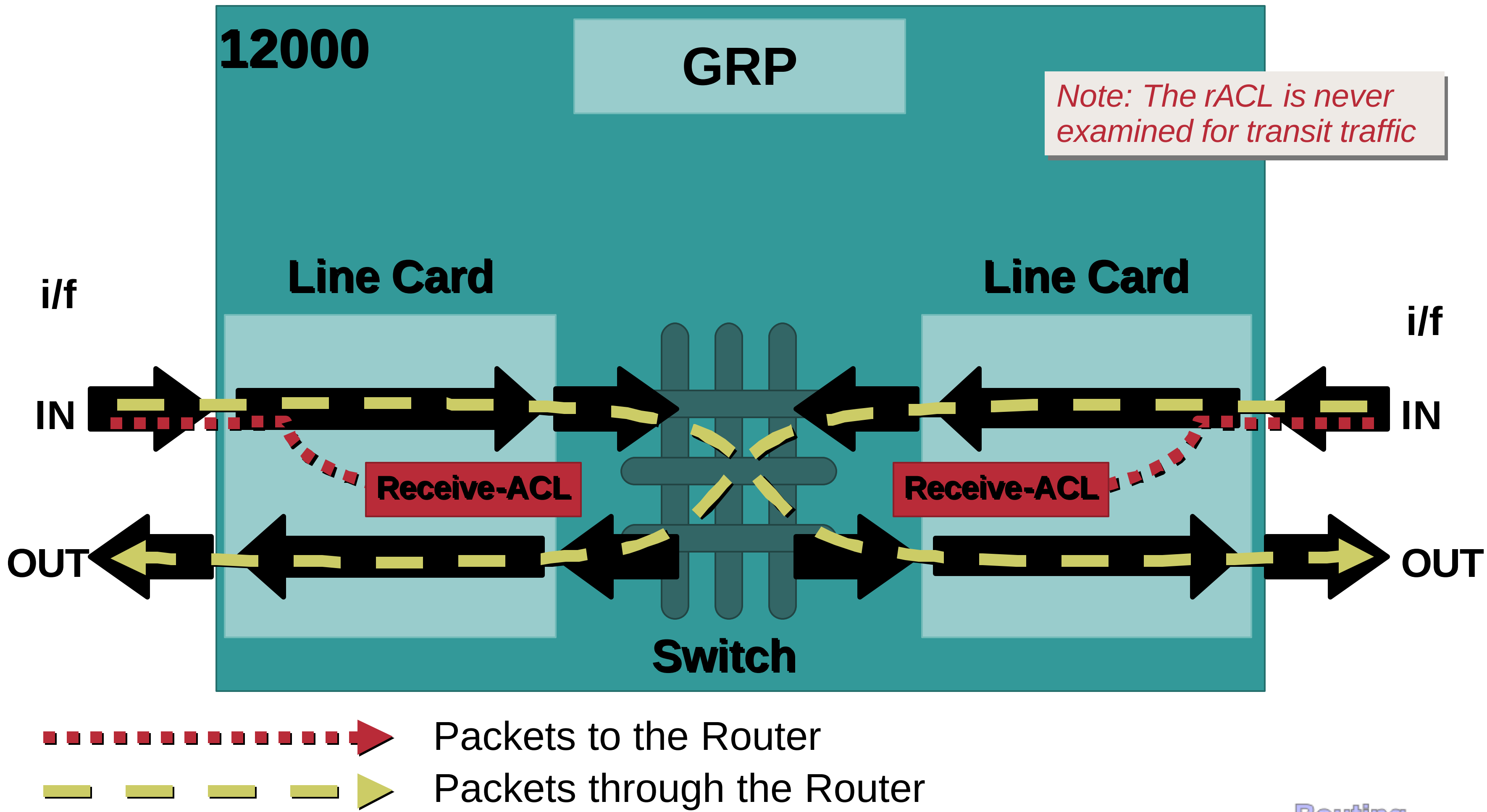
- **CEF entries for traffic destined to router, not through it**
  - Real interface IP addresses
  - Loopback IP addresses

```
12000-1#sh ip cef
Prefix              Next Hop            Interface
10.1.2.0/24         172.16.1.216        GigabitEthernet3/0
10.1.3.0/24         172.16.1.216        GigabitEthernet3/0
172.16.1.196/32     receive
(172.16.1.196 is an interface IP address)
```

- **Packets with next hop receive are sent to the router for processing**
  - Some are handled directly by the LC
  - Others must be sent to the RP (GRP or PRP)
- **Traffic usually routing protocols, management, multicast control traffic**

Receive ACL Traffic Flow

Router(config)# [no] ip receive access-list <num>



12000 rACL Processing

- LC CPU handles rACL processing
- Under attack, LC CPU utilization increases
- Impact depends on LC engine type

  E0/E1/E2: High CPU might impact routing and l2 traffic

  E2 w/ throttle ucode: High CPU → activates throttling, only precedence 6/7 traffic forwarded to RP

  E3: one of 3 queues dedicated for prec. 6/7 traffic, another for L2 keepalives

  E4/E4+: 8 queues, prec. 6/7 and L2 keepalives in dedicated queues

- rACL always improves resiliency to attack

rACLs and Fragments

- **Fragments can be denied via an rACL**
- **Denies fragments and classifies fragment by protocol:**

```
access-list 110 deny tcp any any fragments
access-list 110 deny udp any any fragments
access-list 110 deny icmp any any fragments
```

rACL: Building Your ACL

- Develop list of required protocols
- Develop address requirements
- Determine interface on router

  Does the protocol access 1 interface?

  Many interfaces?

  Loopback or real?

- Deployment is an iterative process

  Start with relatively "open" lists → tighten as needed

rACL: Iterative Deployment

- **Step 1: Identify required protocols via classification ACL**
  - Permit any any for various protocols
  - Get an understanding of what protocols communicate with the router
  - Logging can be used for more detailed analysis
- **Step 2: Review identified packets, begin to filter access to the GRP**
  - Using list developed in step 1, permit only those protocols
  - Deny any any at the end → basic protection AND identify missed protocols

rACL: Iterative Deployment

- **Step 3: Limit source address block**

  Only permit your CIDR block in the source field

  eBGP peers are the exception: they will fall outside CIDR block

- **Step 4: Narrow the rACL permit statements: authorized source addresses**

  Increasingly limit the source addresses to known sources: management stations, NTP peers, etc.

rACL: Iterative Deployment

- **Step 5: Limit the destination addresses on the rACL**
  - **Filter what interfaces are accessible to specific protocols**
  - **Does the protocol access loopbacks only? Real interfaces?**

rACL: Sample Entries

- **OSPF**

```
access-list 110 permit ospf host ospf_neighbour host 224.0.0.5
! DR multicast address, if needed
access-list 110 permit ospf host ospf_neighbour host 224.0.0.6
access-list 110 permit ospf host ospf_neighbour host local_ip
```

- **BGP**

```
access-list 110 permit tcp host bgp_peer host loopback eq bgp
```

- **EIGRP**

```
access-list 110 permit eigrp host eigrp_neighbour host 224.0.0.10
access-list 110 permit eigrp host eigrp_neighbour host local_ip
```

rACL: Sample Entries

- **SSH/Telnet**

```
access-list 110 permit tcp management_addresses host loopback eq 22
access-list 110 permit tcp management_addresses host loopback eq telnet
```

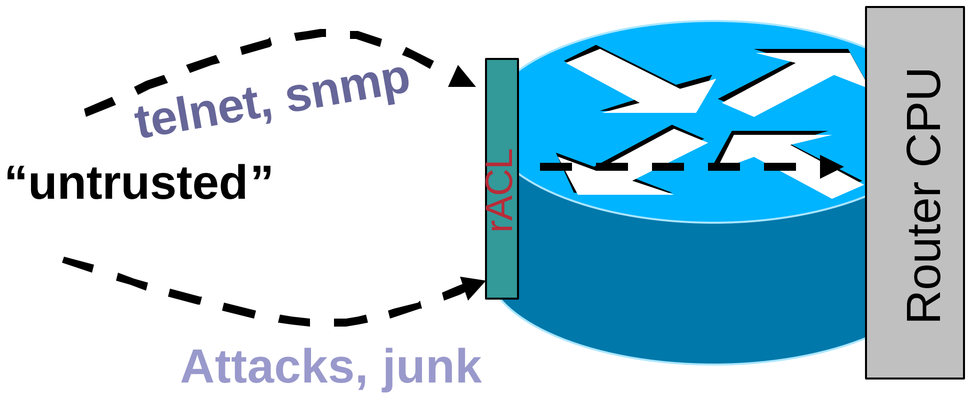
- **SNMP**

```
access-list 110 permit udp host NMS_stations host loopback eq snmp
```

- **Traceroute (router originated)**

```
!Each hop returns a ttl exceeded (type 11, code 3) message and the final destination
returns an ICMP port unreachable (type 3, code 0)
access-list 110 permit icmp any routers_interfaces ttl-exceeded
access-list 110 permit icmp any routers_interfaces port-unreachable
```

Receive ACLs



- **Contain the attack: compartmentalize**
  - **Protect the RP!**
- **Widely deployed and highly effective**
  - **If you have platforms that support rACLs, start planning a deployment**
  - **rACL deployments can easily be migrated to control plane policing (next topic)**
- **Limited platform support**
- **Lack of granularity**

Control Plane Policing (CoPP)

Control Plane Policing (CoPP)

- rACLs are great but
  - Limited platform availability
  - Limited granularity—permit/deny only
- Need to protect all platforms
  - To achieve protection today, need to apply ACL to all interfaces
  - Some platform implementation specifics
- Some packets need to be permitted but at limited rate
  - Think ping :-)

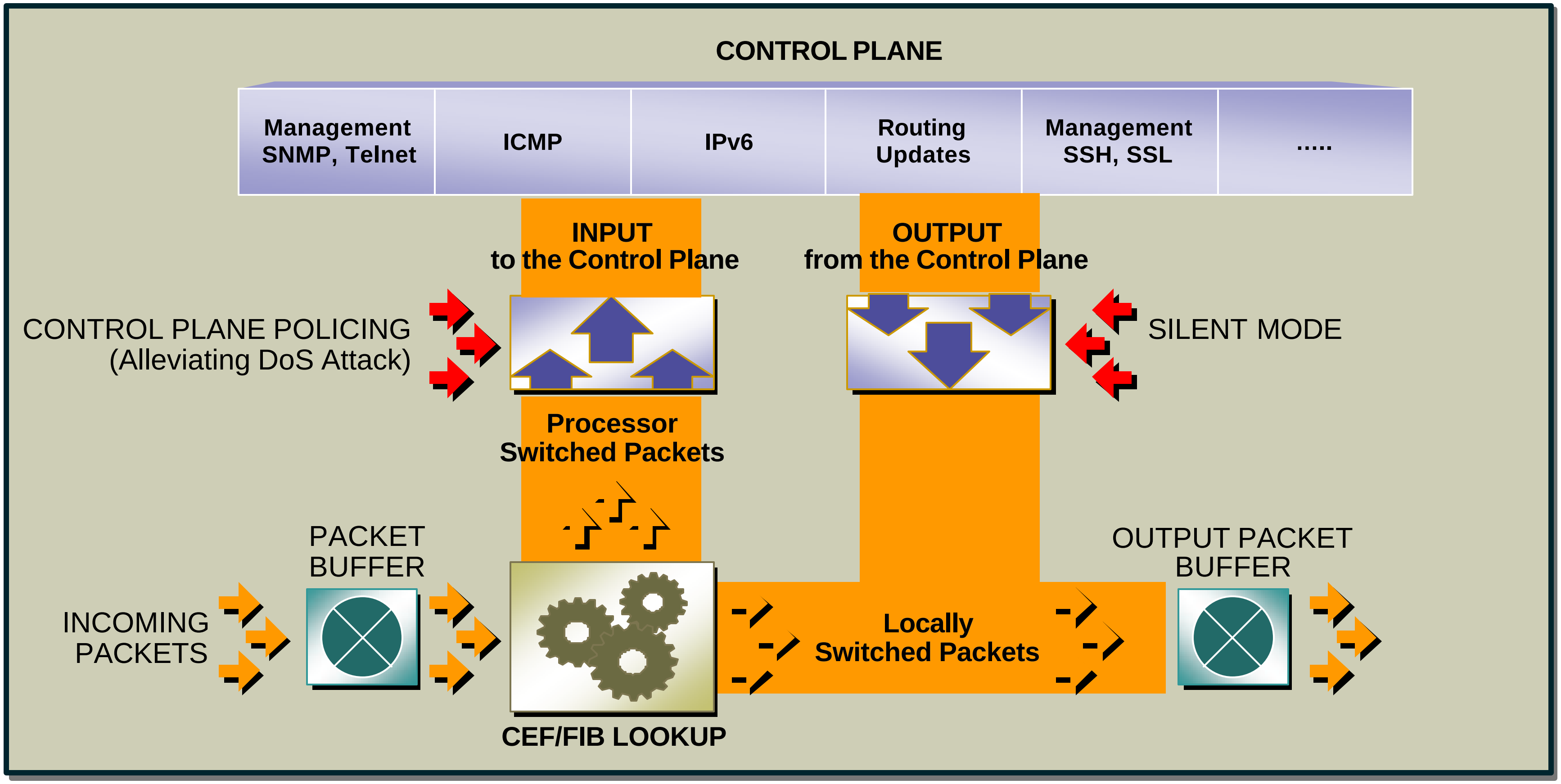
Control Plane Policing (CoPP)

- **CoPP uses the Modular Qos CLI (MQC) for QoS policy definition**
- **Consistent approach on all boxes**
- **Dedicated control-plane "interface"**

  Single point of application
- **Highly flexible: permit, deny, rate limit**
- **Extensible protection**

  Changes to MQC (e.g. ACL keywords) are applicable to CoPP

Control Plane Policing Feature



Configuring CoPP

- **CoPP policy is applied to the control-plane itself**

```
Router(config)# control-plane
Router(config-cp)# service-policy input  control-plane-policy
```

- **Three required steps:**

  Class-map

    Setup class of traffic

  Policy-map

    Define the actual QoS policy: rate limiting and actions

  Apply CoPP policy to control plane "interface"

Deploying CoPP

- Do you know what rate of TCP/179 traffic is normal or acceptable?
- rACL are relatively simple to deploy
    - I know that I need BGP/OSPF/etc., deny all else
- To get the most value from CoPP, detailed planning is required
    - Depends on how you plan to deploy it
    - Bps vs. pps
    - In vs. out

Deploying CoPP

- **One option: mimic rACL behavior**
  - **Apply rACL to a single class in CoPP**
  - **Same limitations as with rACL: permit / deny only**

  Recommendation: Develop multiple classes of control plane traffic
  - **Apply appropriate rate to each**
  - **"Appropriate" will vary based on network, risk tolerance, risk assessment**
- **Flexible class definition allows extension of model**
  - **Fragments, TOS, ARP**

Step 1: Classification

- **Identity traffic destined to routers**
  - **Some is easy (BGP, OSPF, etc.)**
  - **What else?**
- **NetFlow can be used to classify traffic**
  - **Need to export and review**
- **Classification ACL can be used to identify required protocols**
  - **Series of permit statements that provide insight into required protocols**
  - **Initially, many protocols can be permitted, only required ones permitted in next step**
- **Regardless of method, unexpected results should be carefully analyzed → do not permit protocols that you can't explain!**

Step 2: Policy Creation

- **Define Classification Policy**
  - Group IP Traffic types identified in Step 1 into different classes
    - Critical -- Traffic crucial to the operation of the network
    - Important -- Traffic necessary for day-to-day operations
    - Normal -- Traffic expected but not essential for network operations
    - Undesirable -- Explicitly "bad" or "malicious" traffic to be denied access to the RP
    - Default -- All remaining traffic destined to RP that has not been identified
- **Create ACLs to define traffic**
  - Use ACLs *with unique numbers* to represent each class defined above
- **Create Class Maps to collect access-lists**
  - Associate the Traffic Separation ACLs developed above with class-maps with "descriptive" names
    - Use the simple "match access-group <*acl-number*>" format
    - Add the "match protocol" format as necessary (e.g. ARP)
    - Use class-default to identify all unclassified packets

Step 2: Policy Creation

Cisco.com

- **Packet Classifcation**

  The router IP address for control/management traffic is 10.1.1.1

  - **Critical -- ACL 120**
  - **Important -- ACL 121**
  - **Normal -- ACL 122**
  - **Undesirable -- ACL 123**
  - **Default -- No ACL required**

```
! CRITICAL -- Defined as routing protocols
access-list 120 permit tcp host 10.1.1.2 eq bgp host 10.1.1.1 gt 1024
access-list 120 permit tcp host 10.1.1.2 gt 1024 host 10.1.1.1 eq bgp
access-list 120 permit tcp host 10.1.1.3 eq bgp host 10.1.1.1 gt 1024
access-list 120 permit tcp host 10.1.1.3 gt 1024 host 10.1.1.1 eq bgp
access-list 120 permit ospf any host 224.0.0.5
access-list 120 permit ospf any host 224.0.0.6
access-list 120 permit ospf any any
```

```
! IMPORTANT -- Defined as traffic required to access and manage the router
access-list 121 permit tcp host 10.2.1.1 host 10.1.1.1 established
access-list 121 permit tcp 10.2.1.0 0.0.0.255 host 10.1.1.1 range 22 telnet
access-list 121 permit tcp host 10.2.2.1 host 10.1.1.1 eq 443
access-list 121 permit udp host 10.2.2.2 host 10.1.1.1 eq snmp
access-list 121 permit udp host 10.2.2.3 host 10.1.1.1 eq ntp
```

TACACS+ return traffic (arrow pointing to the line `access-list 121 permit tcp host 10.2.1.1 host 10.1.1.1 established`)

Step 2: Policy Creation

- **Packet classifcation (continued)**

  Critical -- ACL 120
  Important -- ACL 121
  Normal -- ACL 122
  Undesirable -- ACL 123
  Default -- No ACL required

```
! NORMAL -- Defined as other traffic destined to the router to track and limit
access-list 122 permit icmp any any ttl-exceeded
access-list 122 permit icmp any any port-unreachable
access-list 122 permit icmp any any echo-reply
access-list 122 permit icmp any any echo
access-list 122 permit icmp any any packet-too-big
```

Allows router originated traceroute (ttl-exceeded / port-unreachable)

```
! UNDESIRABLE -- Defined as traffic explicitly blocked (known malicious)
access-list 123 permit udp any any eq 1434
access-list 123 permit ip any any fragments
```

Use "permit" here because the police action will be "drop/drop" for conform/exceed-actions

Step 2: Classification Policy

- **Create class-maps to complete the traffic-classification process**
  - **Use the access-lists defined on the previous slides to specify which IP packets belong win which classes**
- **Class-maps permit multiple match criteria, and nested class-maps**
  - **match-any requires that packets meet only one "match" criteria to be considered "in the class"**
  - **match-all requires that packets meet all of the "match" criteria to be considered "in the class"**
- **A "match-all" classification scheme with a simple, single-match criteria will satisfy initial deployments**
- **Traffic destined to the "undesirable" class should follow a "match-any" classification scheme**

```
! Define a class for each "type" of traffic and associate the appropriate ACL
class-map match-all CoPP-critical
  match access-group 120
class-map match-all CoPP-important
  match access-group 121
class-map match-all CoPP-normal
  match access-group 122
class-map match-any CoPP-undesirable
  match access-group 123
```

Step 3: Policing Policy

- **Class-maps defined in Step 2 need to be "enforced" by using a policy-map to specify appropriate service policies for each traffic class**

  For example:

  For critical traffic, no policy is specified - critical traffic has unrestricted access to the Route Processor.

  For undesirable traffic types, all actions are unconditionally "drop" regardless of rate

  For important and normal traffic types, all actions are "transmit" to start out

  For default traffic, rate-limit the amount of traffic permitted above a certain bps

  Note: all traffic that fails to meet the matching criteria belongs to the default traffic class, which is user configurable, but cannot be deleted

```
! Example "Baseline" service policy for each traffic classification
policy-map CoPP
  class CoPP-critical
    police 8000 1500 1500 conform-action transmit exceed-action transmit
  class CoPP-undesirable
    police 8000 1500 1500 conform-action drop exceed-action drop
    <or simply>
    drop
  class CoPP-important
    police 125000 1500 1500 conform-action transmit exceed-action transmit
  class CoPP-normal
    police 15000 1500 1500 conform-action transmit exceed-action transmit
  class class-default
    police 8000 1500 1500 conform-action transmit exceed-action drop
```

Step 4: Apply Policy to “Interface”

- **Apply the policy-map created in Step 3 to the “Control Plane”**
  - **The new global configuration CLI “control-plane” command is used to enter “control-plane configuration mode”**
  - **Once in control-plane configuration mode, attach the service policy to the control plane in either the “input” or “output” direction**
    - **Input -- Applies the specified service policy to packets that are entering the control plane**
    - **Output -- Applies the specified service policy to packets that are exiting the control plane**
  - **A service policy may be applied to the control plane in one or both directions (two separate statements**)

Centralized CoPP:

Router(config)# control-plane

Router(config-cp)# service-policy [input | output] <*policy-map-name*>

Distributed CoPP (DCoPP):

Router(config)#control-plane slot <n>

Router(config-cp)#service-policy input control-plane-in <policy-map-name>

```
! Example
! This applies the policy-map to the Control Plane
control-plane
  service-policy input CoPP-In
```

CoPP and IP Fragments

Cisco.com

- **Under normal circumstances, routers should *never* see fragmented IP packets in the receive path**

  Some exceptions may apply if tunneling (GRE and/or IPsec, e.g.) is involved…

- **Filtering fragments using CoPP**

  The most effective approach is to create a new class, e.g. *fragments* and associate a drop policy with this class.

  The use of a drop policy for fragments within a CoPP policy will deny all non-initial fragments from accessing the router (and remember to use a "permit" match)

  If needed, changes can be made to use a rate limited policy instead.

- **Order is important**

  The order of defining classes within a policy is important

  - Packets will be classified to a particular class based on the order that they are called within a policy.

Control Plane Policing (CoPP) Deployment
Fragmented Packets…


```
! Define policy for IP fragments -- "permit" means they'll be dropped!
access-list 110 permit tcp any any fragments
access-list 110 permit udp any any fragments
access-list 110 permit icmp any any fragments
access-list 110 permit ip any any fragments
```

```
! Associate ACL's with Class Maps
class-map cpp-fragments
   match access-group 110
class-map cpp-critical
   match access-group 120
```

```
! Define policy for IP fragments -- "permit" means they'll be dropped!
! Order is important! -- must drop fragments before any other policies
policy-map cpp
  class cpp-fragments
    police cir 8000 conform-action transmit exceed-action drop
    ! if the unconditional packet drop command is supported, you can configure drop
  class cpp-critical
    ! no operation specified - this cla[illegible] unrestricted access to the Route Processor
```

Alternate policy for IP fragments. If conditions dictate, use a rate limiting policy!

Monitoring CoPP

- **"Show" commands to review service-policy transmit and drop rates to ensure that the appropriate traffic types and rates are receiving the appropriate policing policy**
  - "show access-list" displays hit counts on a per ACL entry (ACE) basis
  - The presence or absence of hits indicates flows (or lack there of) for that data type to the control plane as expected
  - Large numbers of packets or an unusually rapid rate increase in packets processed may be suspicious and should be investigated
  - Lack of packets may also indicate unusual behavior or that a rule may need to be rewritten
  - "show policy-map control-plane" is invaluable for reviewing and tuning site-specific policies and troubleshooting CoPP
  - Displays dynamic information about number of packets (and bytes) conforming or exceeding each policy definition
  - Useful for ensuring that appropriate traffic types and rates are reaching the route processor
- **Use SNMP queries to automate the process of reviewing service-policy transmit and drop rates**
  - The Cisco QoS MIB (CISCO-CLASS-BASED-QOS-MIB) provides the primary mechanisms for MQC-based policy monitoring via SNMP

Show Policy-map Command


```
Router#show policy-map control-plane input
 Control Plane

  Service-policy input: CoPP

    Class-map: Classify (match-all)
      16 packets, 2138 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: access-group 120
      police:
          cir 125000 bps, bc 1500 bytes
        conformed 16 packets, 2138 bytes; actions:
          transmit
        exceeded 0 packets, 0 bytes; actions:
          transmit
        conformed 0 bps, exceed 0 bps

    Class-map: class-default (match-any)
      250 packets, 84250 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: any
      police:
          cir 8000 bps, bc 1500 bytes
        conformed 41 packets, 5232 bytes; actions:
          transmit
        exceeded 0 packets, 0 bytes; actions:
          drop
        conformed 0 bps, exceed 0 bps
Router#
```

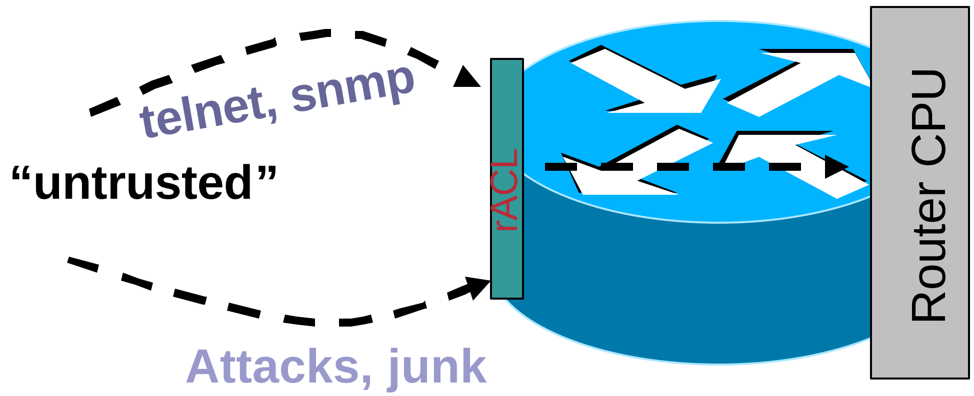
- Service Policy Map name and “direction”
- Class-map name and “criteria”
- Number of packets/bytes matched
- ACL name/number
- police “action”
- Default class
- police “action”

CoPP and SNMP


```
! Using SNMP...
[Linux]$ snmpwalk -m all 10.82.69.157 ciSco .1.3.6.1.4.1.9.9.166.1.15.1.1.2
enterprises.cisco.ciscoMgmt.ciscoCBQosMIB.ciscoCBQosMIBObjects.cbQosClassMapStats.cbQosCMStatsTabl
e.cbQosCMStatsEntry.cbQosCMPrePolicyPkt.1035.1037 = Counter32: 3924
[Linux]$ snmpwalk -m all 10.82.69.157 ciSco .1.3.6.1.4.1.9.9.166.1.15.1.1.5
enterprises.cisco.ciscoMgmt.ciscoCBQosMIB.ciscoCBQosMIBObjects.cbQosClassMapStats.cbQosCMStatsTabl
e.cbQosCMStatsEntry.cbQosCMPrePolicyByte.1035.1037 = Counter32: 344523
[Linux]$

! Via CLI...
Router#sh pol control-plane
 Control Plane
  Service-policy input: Classify
    Class-map: class-default (match-any)
      3924 packets, 344523 bytes
      5 minute offered rate 1000 bps, drop rate 0 bps
      Match: any
      police:
          cir 12500 bps, bc 1500 bytes
        conformed 3875 packets, 336178 bytes; actions:
          transmit
        exceeded 49 packets, 8345 bytes; actions:
          drop
        conformed 1000 bps, exceed 0 bps
Router#
```

Control Plane Policing



- **Superset of rACL: start planning your migrations**
- **Provides a cross-platform methodology for protecting the control plane**

  Consistent "show" command and MIB support
- **Granular: permit, deny and rate-limit**
- **Default-class provides flexibility**
- **Platform specifics details: centralized vs. distributed vs. hardware**

Agenda

- Infrastructure security overview
- Preparing The Network
- Router Security: A Plane Perspective
- Tools and Techniques
- Platform Architecture
- Conclusions

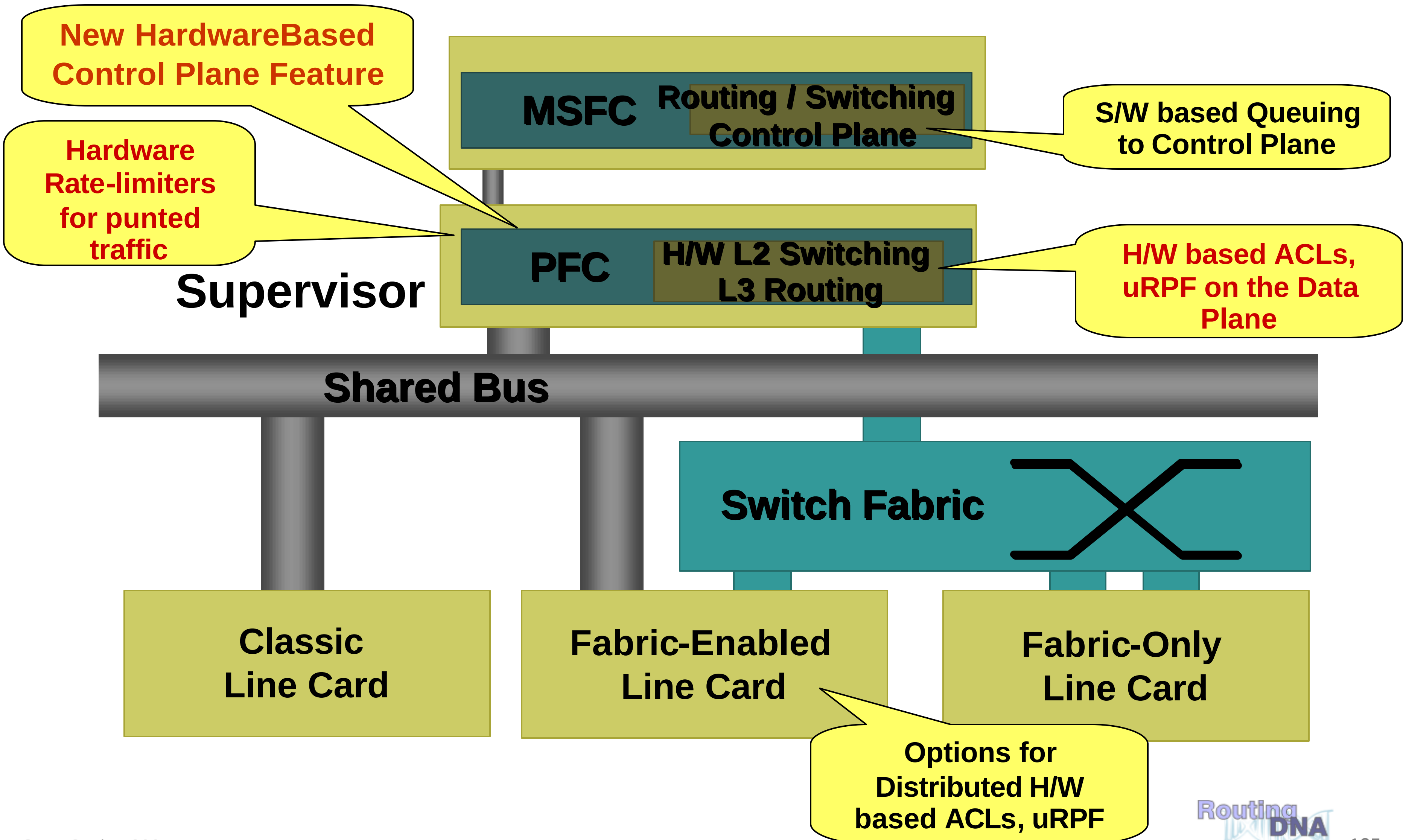
Router Architecture

Cisco.com

- In addition to best practices and feature, hardware architecture plays an important role in protecting devices
  - Defense in depth
  - Utilize various techniques to compartmentalize router components
- Understand your platform design to truly understand risk
  - We will review core platforms: 6500/7600 and 12000

Cisco 7600: Sup720 DoS Protection

Cisco.com



Introduction – What Hardware-based CPU Rate Limiters Are Available?

Cisco.com

Unicast Rate Limiters	
CEF Receive	Traffic destined to the Router
CEF Glean	ARP packets
CEF No Route	Packets with not route in the FIB
IP Errors	Packets with IP checksum or length errors
ICMP Redirect	Packets that require ICMP redirects
ICMP No Route	ICMP unreachables for unroutable packets
ICMP ACL Drop	ICMP unreachables for admin deny packets
RPF Failure	Packets that fail uRPF check
L3 Security	CBAC, Auth-Proxy, and IPSEC traffic
ACL Input	NAT, TCP Int, Reflexive ACLs, Log on ACLs
ACL Output	NAT, TCP Int, Reflexive ACLs, Log on ACLs
VACL Logging	CLI notification of VACL denied packets
IP Options	Unicast traffic with IP Options set (B/BXL)
Capture	Used with Optimized ACL Logging

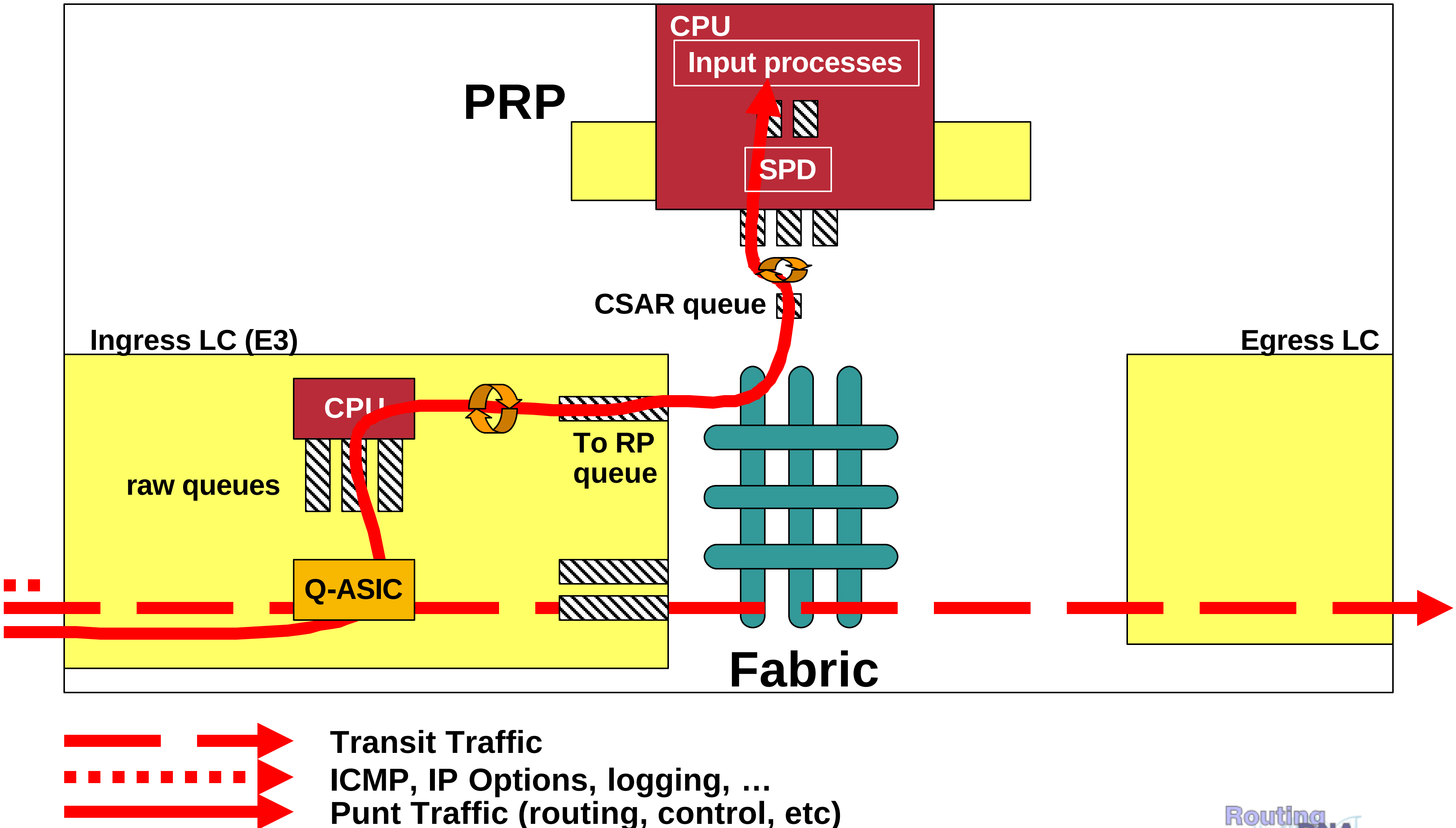
Multicast Rate Limiters	
Multicast FIB-Miss	Packets with no mroute in the FIB
IGMP	IGMP packets
Partial Shortcut	Partial shortcut entries
Directly Connected	Local multicast on connected interface
IP Options	Multicast traffic with IP Options set (B/BXL)
V6 Directly Connect	Packets with no mroute in the FIB
V6*, G M Bridge	IGMP Packets
V6*, G Bridge	Partial shortcut entries
V6 S, G Bridge	Partial shortcut entries
V6 Route Control	Partial shortcut entries
V6 Default Route	Multicast traffic with IP Options set
V6 Second Drop	Multicast traffic with IP Options set

Shared across the 10 hardware Revocation Lists.

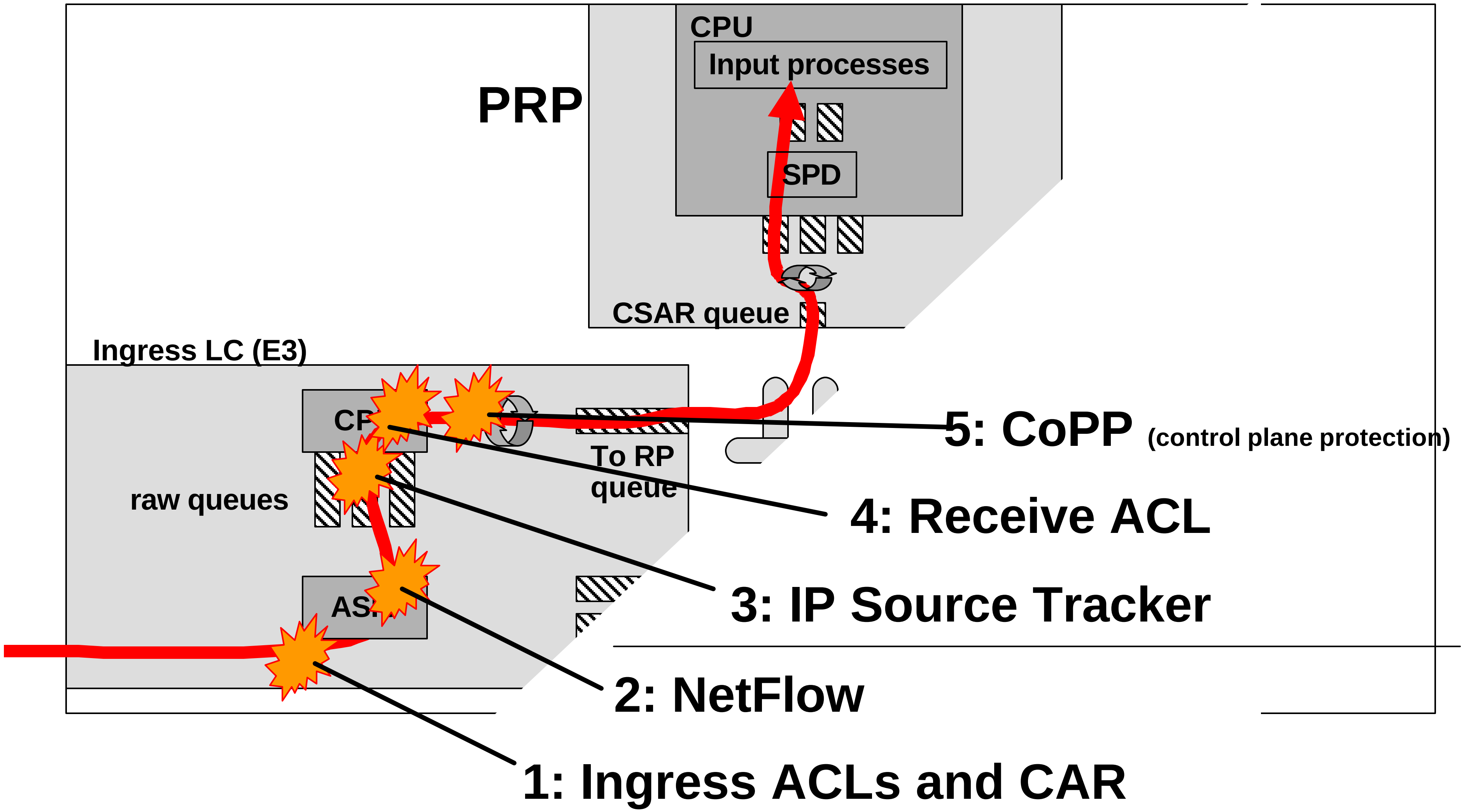
Layer 2 Rate Limiters	
L2PT	L2PT encapsulation/decapsulation
PDU	Layer 2 PDUs

General Rate Limiters	
MTU Failure	Packets requiring fragmentation
TTL Failure	Packets with TTL<=1

Data Paths in the Cisco 12000



12000 Feature Ordering



Router Architecture and Infrastructure Security

- "Build it into the hardware"
- Risk analysis requires platform architecture understanding
  - How does a particular platform handles data, control and management plane packets?
  - What about transitions between planes?
- Develop your protection schemes with this data in hand

Agenda

- Infrastructure security overview
- Preparing The Network
- Router Security: A Plane Perspective
- Tools and Techniques
- Platform Architecture
- Conclusions

Summary

- **Understand the risk**

  Your infrastructure needs to be protected from direct and indirect attacks

- **Want to deploy voice? Want to deploy video? Want to deploy xyz?**

  All services deployment depend on an available infrastructure

- **Understand the techniques/features and apply them appropriately**

  Edge filters: iACLs

  Control plane traffic filtering: rACL

  Next-phase of control plane filtering (including policing): CoPP

- **Each feature has pros/cons**

  Ultimately, mix and match as needed: remember defense in depth

Summary

Cisco.com

- **Take infrastructure protection into account in network design**
  - **Key component of network availability**
- **Review your current protection schemes**
  - **Identify gaps and areas of exposure**
  - **Develop a plan for protection**
- **Start planning you deployments!**
  - **Can be difficult but certainly worthwhile!**
  - **Many customers have widespread deployments and have seen the benefits**

Interesting Links

- **iACL deployment guide**

  http://www.cisco.com/warp/public/707/iacl.html

- **rACL deployment guide**

  http://www.cisco.com/warp/public/707/racl.html

- **CoPP deployment guide**

  http://www.cisco.com/en/US/products/sw/iosswrel/ps1838/products_white_paper09186a0080211f39.shtml

- **Cisco Network Foundation Protection (NFP)**

  http://www.cisco.com/warp/public/732/Tech/security/infrastructure/

- **SP security archive**

  ftp://ftp-eng.cisco.com/cons/isp/security/

- **NANOG**

  http://www.nanog.org/previous.html

  http://www.nanog.org/ispsecurity.html

Complete Your Online Session Evaluation!

Cisco.com

Por favor, complete el formulario de evaluación.

Muchas gracias.

Session ID: SEC-2101

Network Core Infrastructure Protection: Best Practices

CISCO SYSTEMS