Router Security and Infrastructure Protection

Yusuf Bhaiji

Cisco Systems

Agenda

- Introduction to Core Security

  Denial of Service (DoS) and Worm Review

  Six-Phase Methodology

- Infrastructure Security

  RFC 2827/BCP 38

  Infrastructure ACLs

  Flexible Packet Matching

- Network Telemetry

  SNMP, RMON and Their Ilk

  NetFlow for Security Purposes

Agenda (Cont.)

- Traceback Techniques
  - NetFlow Traceback Techniques
  - Attract and Analyze: Sinkholes
- Reacting to Attacks
  - Reacting with ACL
  - Reacting with BGP
  - Packet Scrubbing

Simple Methodology

- Simple methodology—expanding the scope

  Best practices to:

    Protect the device

    Protect the infrastructure

- With a solid foundation in place, we turn our attention to leveraging the network itself as a security toolkit

Denial of Service (DoS) and Worm Review

What Is Core Security?

- Often thought of as “SP Security”
  - What is an SP today?
- Internal networks are no longer truly internal
  - Tunneling
  - VPN
  - Worms, worms, worms
- The infrastructure is critical; if we can’t protect it, nothing else matters
  - Edge security initiatives abound: NAC, 802.1X, HIPS (CSA), personal firewalls, etc.

Denial of Service Attacks

- We understand intrusions (patch, patch, patch ;-))
- What about DoS? Do “the right things” and still suffer
- The vast majority of modern DoS attacks are distributed

  DDos IS DoS

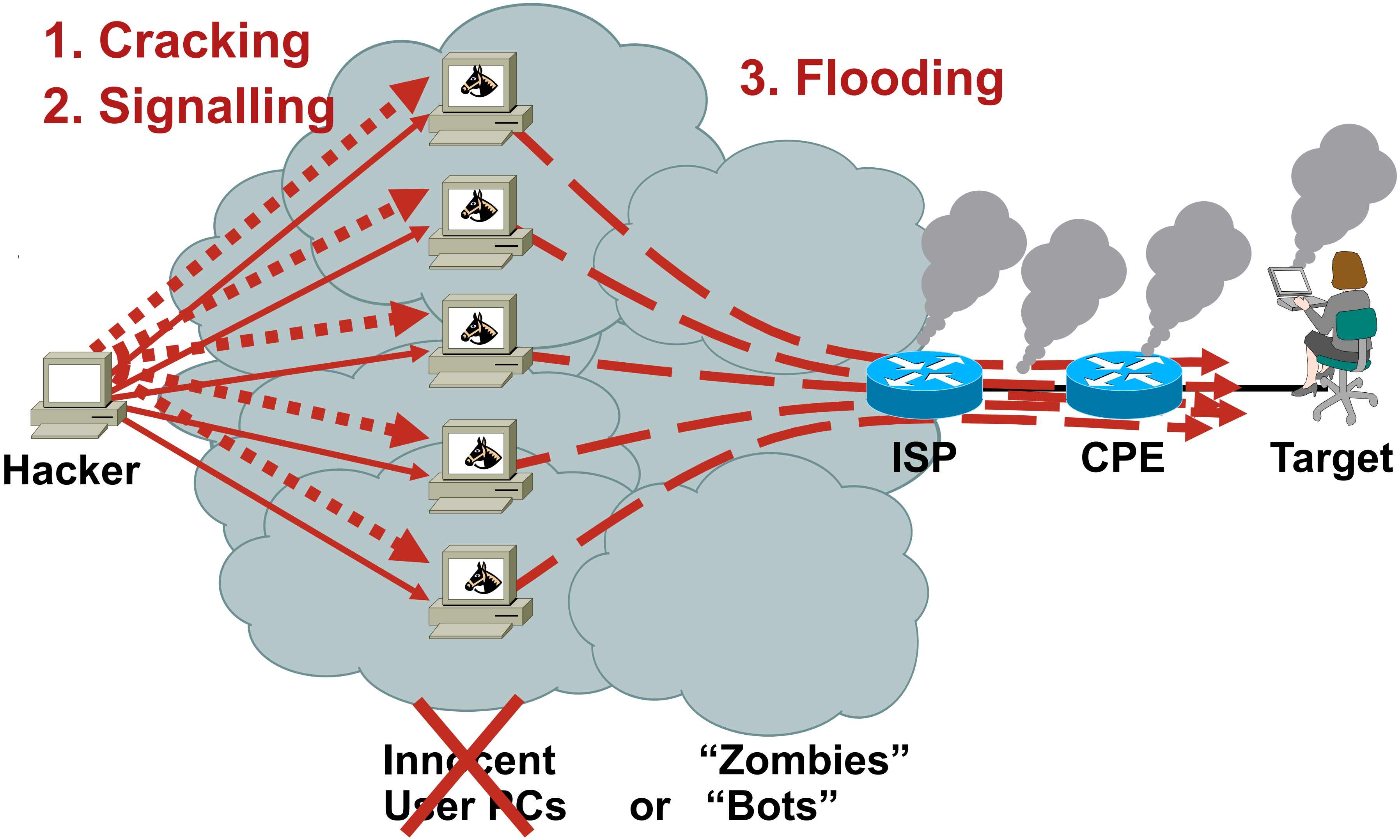
- DoS is often driven by financial motivation

  DoS for hire :-(

  Economically-driven miscreant community

- DoS cannot be ignored; your business depends on effective handling of attacks

DoS: The Procedure



1. Cracking
2. Signalling
3. Flooding

Hacker

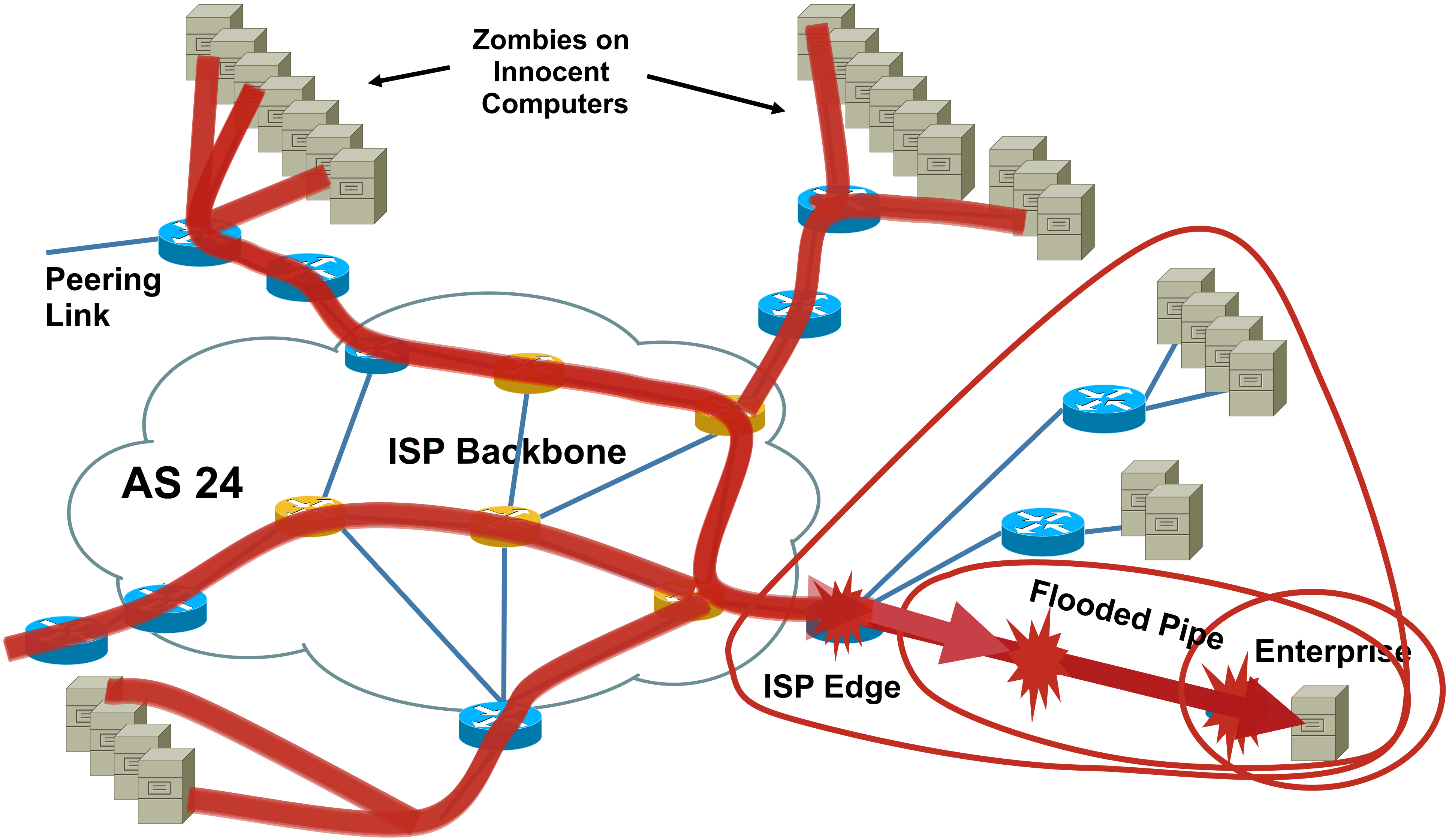
ISP

CPE

Target

~~Innocent User PCs~~ or "Zombies" "Bots"

An SP View: Denial of Service



Denial of Service Trends

- Multipath

  Truly distributed

  DNS servers, large botnets

- Multivector

  SYN AND UDP AND…

- Use of non-TCP/UDP/ICMP protocols

  Get past ACLs

  Increased awareness in community

- Financial incentive

  SPAM, DoS-for-hire

  Large, thriving business

  Forces us to reassess the risk profile

Infrastructure Attacks

- Infrastructure attacks increasing in volume and sophistication

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

  Presentations about routers, routing and Cisco 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 valuable in 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 UPnP 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.)

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
  - TTL and other header variances
- At the core SP level, the aggregate affects of a worm can be substantial
- Worm severity is escalating and evolving

Botnets

- Botnet: a collection of compromised machines running programs under a common command and control infrastructure
- Building the Botnet:

  Viruses, worms; infected spam; drive-by downloads; etc.

- Controlling the Botnet:

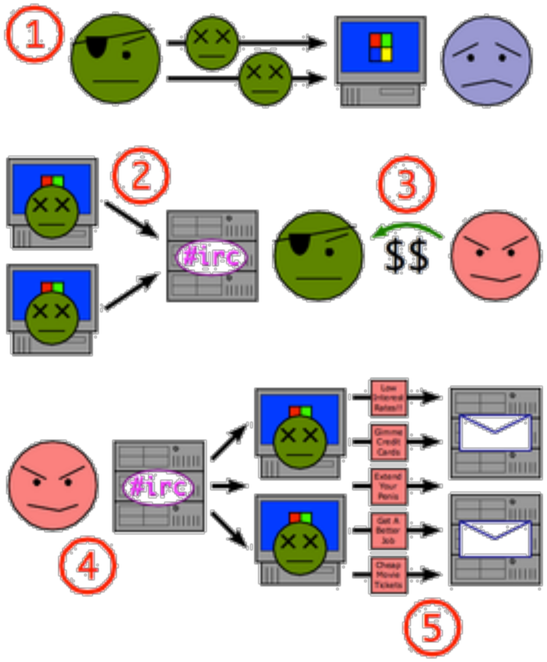
  Covert-channel of some form; typically IRC

  Historically have used free DNS hosting services to point bots to the IRC server

  Recent attempts to sever the command infrastructure of botnets has resulted in more sophisticated control systems

  Control services increasingly placed on compromised high-speed machines (e.g., in academic institutions)

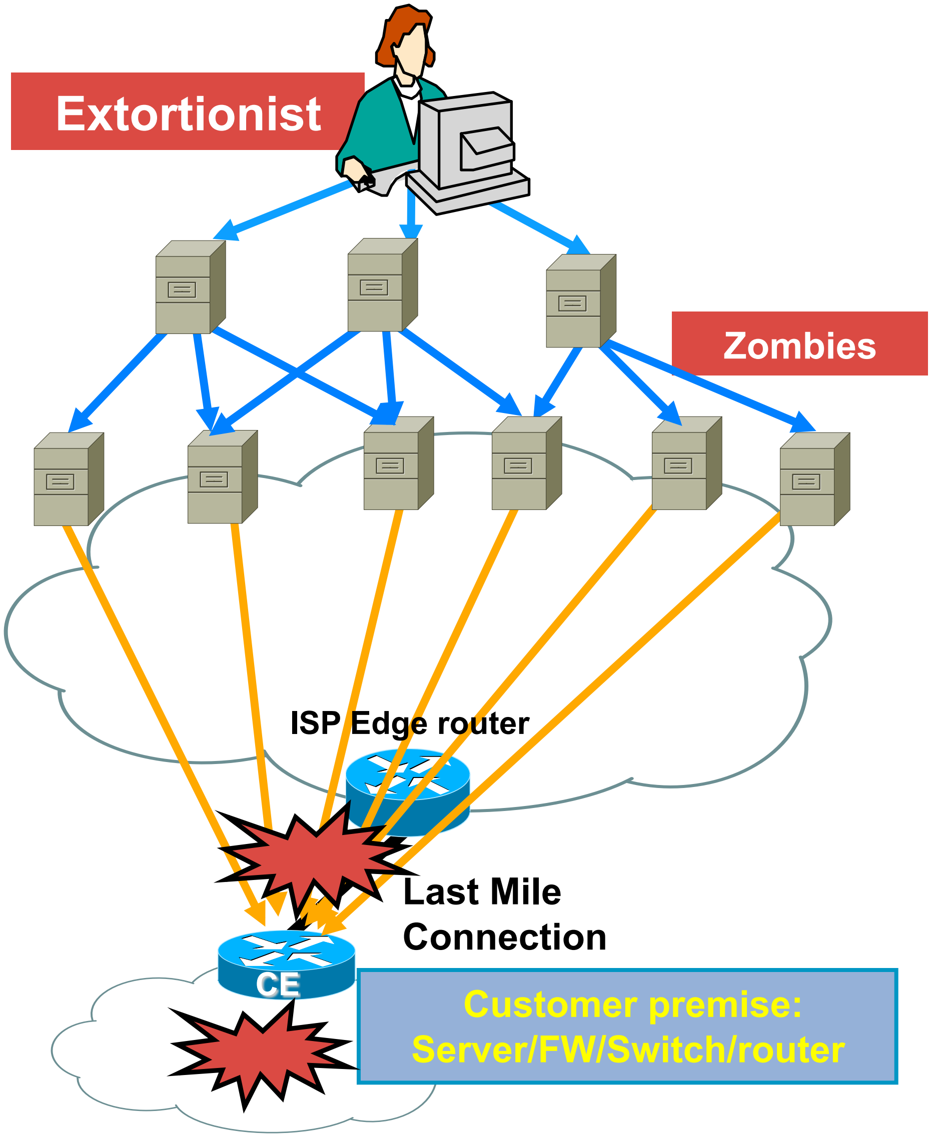
  Redundant systems and blind connects are implemented for resiliency



Using a Botnet to Send Spam

1. A botnet operator sends out viruses or worms, infecting ordinary users' Windows PCs
2. The PCs log into an IRC server or other communications medium
3. A spammer purchases access to the botnet from the operator
4. The spammer sends instructions via the IRC server to the infected PCs
5. ...causing them to send out spam messages to mail servers

Botnets Make DDoS Attacks Easy



- Botnets for Rent!
- A “Botnet” is a group of compromised computers on which extortionists have installed special programs (zombies) that can be directed to launch DoS attacks against a specific target.

  Botnets are triggered from a “central controller”

  Botnets allow for all the types of DDOS attacks: ICMP Attacks, TCP Attacks, UDP Attacks, HTTP overload

  Options for deploying Botnets are extensive and new tools are created to exploit the latest system vulnerabilities
- A relatively small Botnet can cause a great deal of damage.

  1000 home PCs with an average upstream bandwidth of 128KBit/s can offer more than 100MBit/s against a target
- The size of the attacks are ever increasing and independent of last mile bandwidth

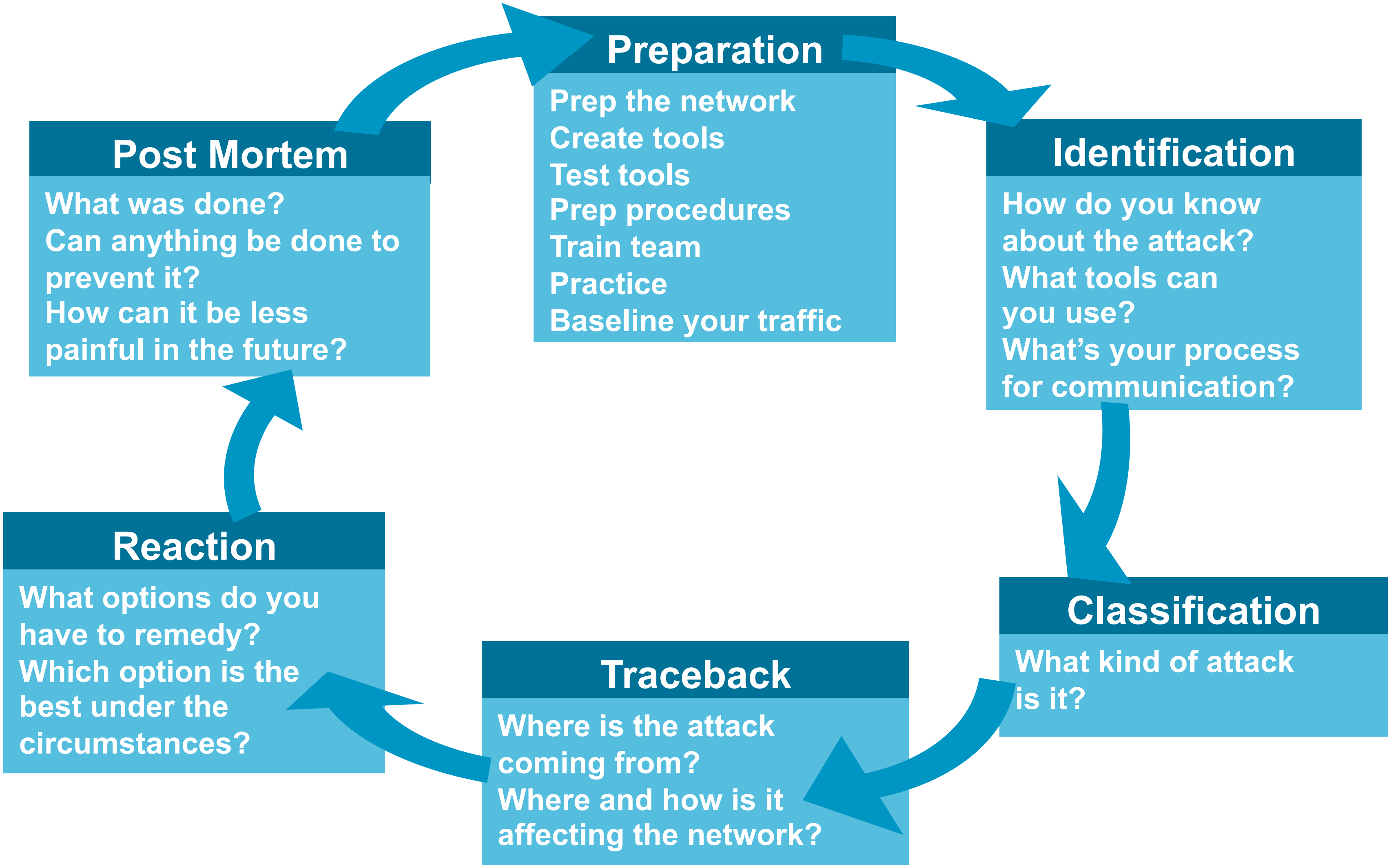
How Do You Respond?

With Money Being the Key Driver of Miscreant Activity, Large Network Operators Need to Respond

- BCP deployment
- Execution of a broad and deep security toolkit
- Rethink some network/service architectures
- Create, staff, and train an operational security (OPSEC) team
- Practice, practice, practice

Six-Phase Methodology

Six Phases of Incident Response



Preparation

Prep the network
Create tools
Test tools
Prep procedures
Train team
Practice
Baseline your traffic

Identification

How do you know about the attack?
What tools can you use?
What's your process for communication?

Classification

What kind of attack is it?

Traceback

Where is the attack coming from?
Where and how is it affecting the network?

Reaction

What options do you have to remedy?
Which option is the best under the circumstances?

Post Mortem

What was done?
Can anything be done to prevent it?
How can it be less painful in the future?

Preparation

Preparation—Develop and Deploy a
Solid Security Foundation

- Includes technical and non-technical components
- Encompasses best practices
- The hardest, yet most important phase
- Without adequate preparation, you are destined to fail
- The midst of a large attack is not the time to be implementing foundational best practices and processes

Preparation

- Know the enemy
  - Understand what drives the miscreants
  - Understand their techniques
- Create the security team and plan
  - Who handles security during an event?
    Is it the security folks? The networking folks?
- Harden the devices
- Prepare the tools
  - Network telemetry
  - Reaction tools
  - Understand performance characteristics

Identification

Identification—How Do You Know You or Your Customer Is Under Attack?

- It is more than just waiting for your customers to scream or your network to crash
- What tools are available?
- What can you do today on a tight budget?

Identification—Ways to Detect

- Customer call

  “The Internet is down”

- Unexplained changes in network baseline

  SNMP: line/CPU overload, drops

  Bandwidth

  NetFlow

- ACLs with logging
- Backscatter
- Packet capture
- Network IPS
- Anomaly detection

Identification—Network Baselines

- NMS baselines
- Unexplained changes in link utilization

  Worms can generate a lot of traffic, sudden changes in link utilization can indicate a worm

- Unexplained changes in CPU utilization

  Worm scans can affect routers/switches resulting in increased CPU - process and interrupt switched traffic

- Unexplained syslog entries
- These are examples

  Changes don't always indicate a security event

  Must know what's normal in order to identify abnormal behavior

Classification

- Classification—understand the details and scope of the attack
    - Identification is not sufficient; once an attack is identified, details matter
    - Guides subsequent actions
- Identification and classification are often simultaneous

Classification

- Qualify and quantify the attack without jeopardizing services availability (e.g., crashing a router)

    What type of attack has been identified?

    What's the effect of the attack on the victim(s)?

    What next steps are required (if any)?

- At the very least:

    Source and destination address

    Protocol information

    Port information

Traceback

- Traceback—what are the sources of the attack?

  How to trace to network ingress points

  Your Internet connection is not the only vector

  Understand your topology

- Traceback to network perimeter

  NetFlow

  Backscatter

  Packet accounting

Traceback

- Retain attack data

    Use to correlate interdomain traceback

    Required for prosecution

    Deters future attacks

    Clarify billing and other disputes

    Post mortem analysis

Reaction

Reaction—Do Something to Counter the Attack

- Should you mitigate the attack?

    Where? How?

- No reaction is a valid form of reaction in certain circumstances
- Reaction often entails more than just throwing an ACL onto a router

Post Mortem

Post Mortem—Analyze the Event

- The step everyone forgets
- What worked? What didn’t? How can we improve?
- Protect against repeat occurrences?
- Was the DoS attack you handled the real threat? Or was it a smoke screen for something else that just happened?
- What can you do to make it faster, easier, less painful in the future?
- Metrics are important

  Resources, headcount, etc.

Routing Protocol Security

Routing Protocol Security

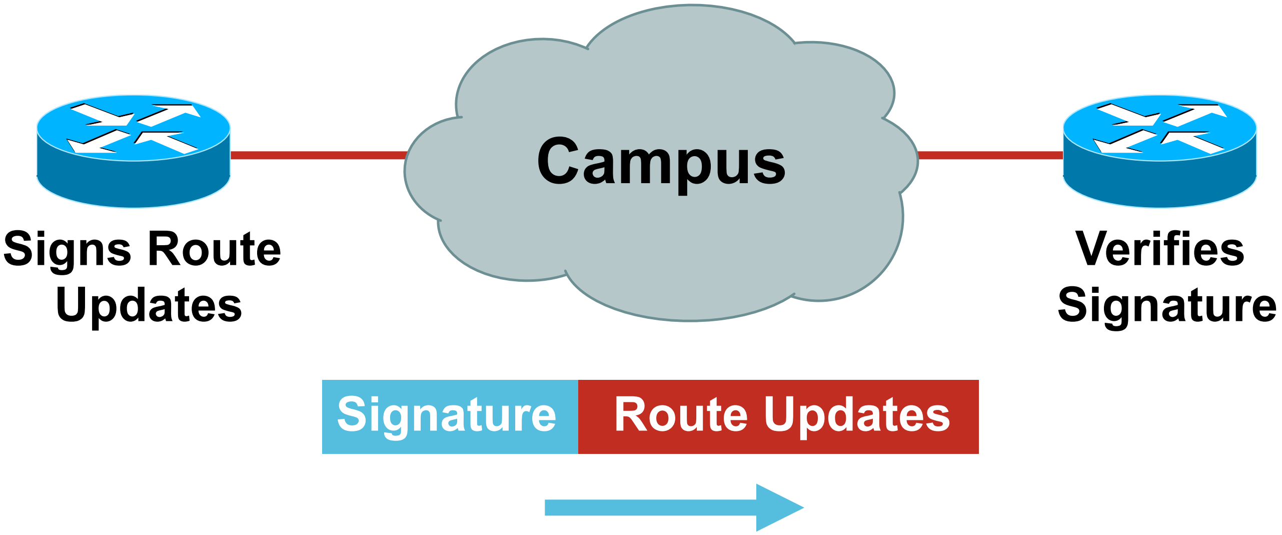
Routing Protocols Can Be Attacked

- Denial of service
- Smokescreens
- False information
- Reroute packets

May Be Accidental or Intentional

Secure Routing—Route Authentication

Configure Routing Authentication



Certifies Authenticity of Neighbor and Integrity of Route Updates

Route Authentication

- Shared key included in routing updates
  - Plain text—protects against accidental problems only
  - Message Digest 5 (MD5)—protects against accidental and intentional problems
- Multiple keys supported
- Supported for BGP, IS-IS, OSPF, RIPv2, and EIGRP
- Update keys before protocol timeout to avoid session bounce
- Often non-implemented
  - “Never seen an attack”
  - “My peer doesn’t use it”

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 4.1.2.1 remote-as 300
 neighbor 4.1.2.1 description Link to Excalibur
 neighbor 4.1.2.1 send-community
 neighbor 4.1.2.1 version 4
 neighbor 4.1.2.1 soft-reconfiguration inbound
 neighbor 4.1.2.1 route-map Community1 out
 neighbor 4.1.2.1 password 7 q23dc%$#ert
```

BGP Route Authentication

- Works per neighbor or for an entire peer-group
- Two routers with password mismatch:

  %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

BGP Support for TTL Security Check

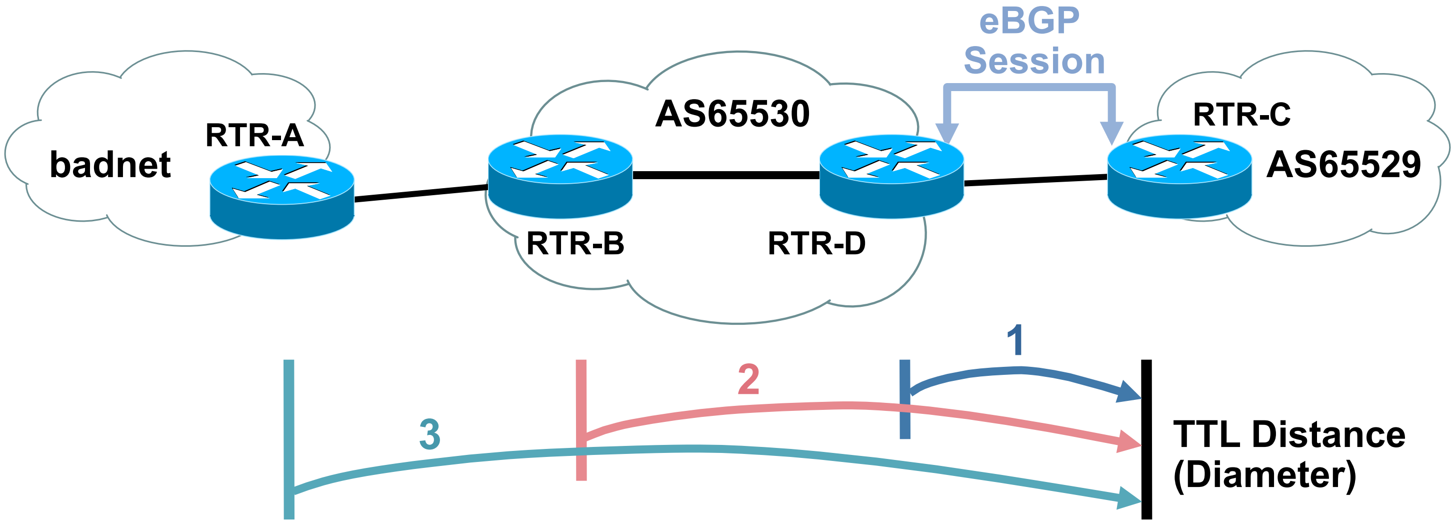
- AKA BGP TTL Security Hack (BTSH)
- Protects eBGP sessions from CPU attacks using forged IP packets
- Prevents attempts to hijack eBGP session by attacker not part of either BGP network or that is not between the eBGP peers
- Configure minimum Time To Live (TTL) for incoming IP packets from a specific eBGP peer

  BGP session established and maintained only if TTL in IP packet header is equal to or greater than configured TTL value. Initial TTL set to 255

  If value is less than configured value packet is silently discarded and no ICMP message generated
- Not supported for iBGP and occurs after MD5 check if enabled
- Available in 12.0(27)S, 12.3(7)T, 12.2(25)S, 12.2(18)SXE

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

BGP TTL Security Check: How Does It Work?



Example on RTR-C:

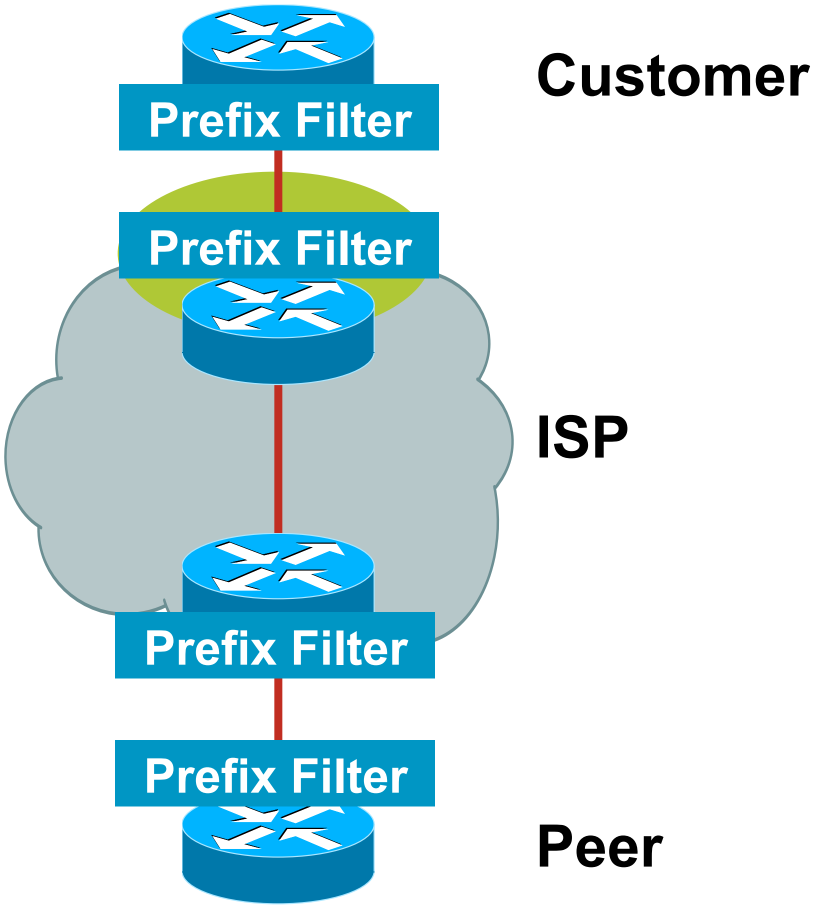
```
router bgp 65529
  neighbor 10.1.1.1 ttl-security hops 1
  ! expected TTL value in IP header is now 254 not 0
```

- Spoofed IP packets may have correct IP source and destination addresses (and TCP source and destination ports); however, unless these packets originate on a network segment that is between the eBGP peers, the TTL values will be less than the "minimum" configured in the BGP TTL security check

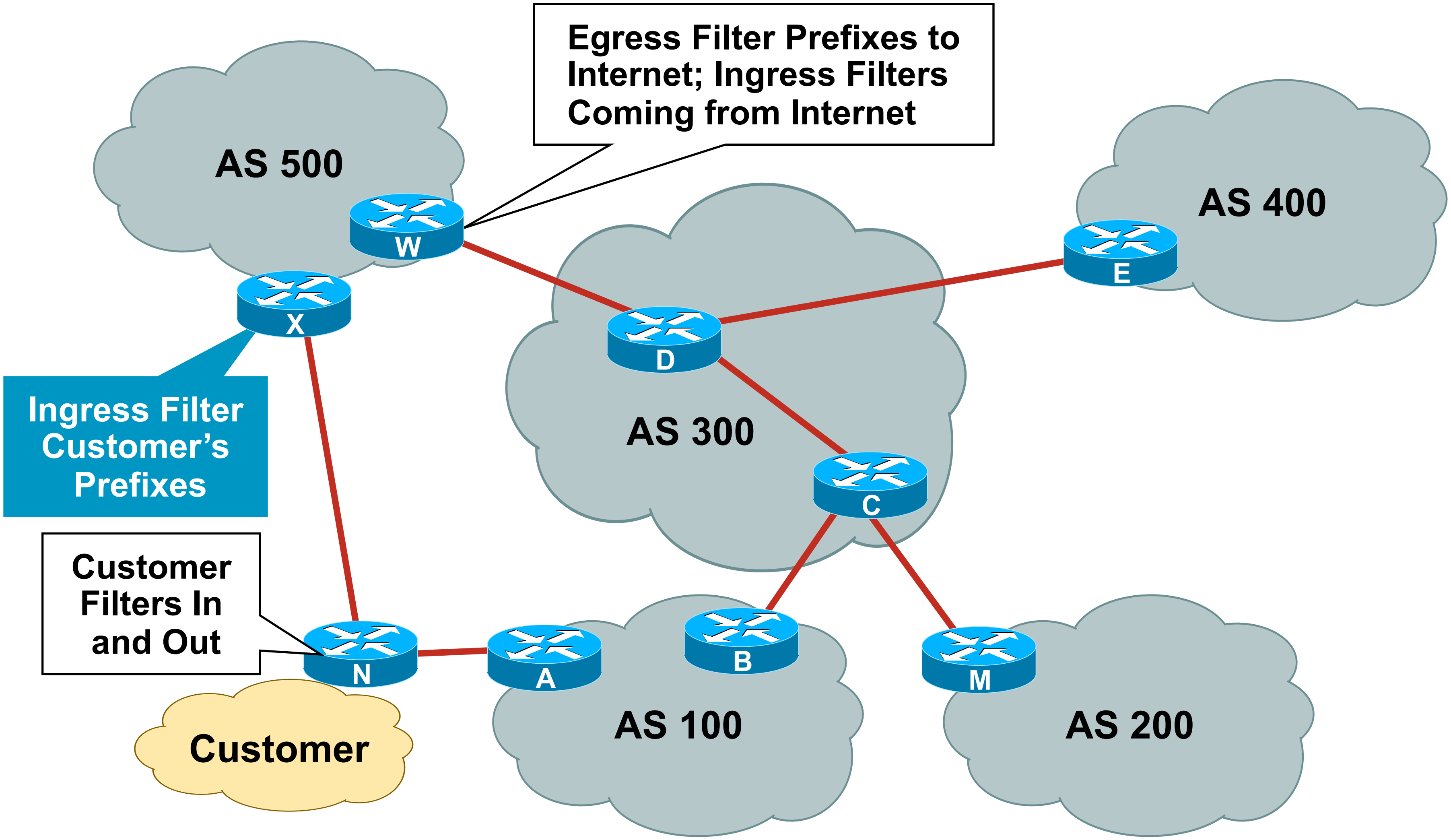
Prefix Filters

Apply Prefix Filters to All eBGP Neighbors to Prevent Injection of False Routing Information

- To/from customers
- To/from peers
- To/from upstreams



Where to Prefix Filter?



Egress Filter Prefixes to Internet; Ingress Filters Coming from Internet

Ingress Filter Customer’s Prefixes

Customer Filters In and Out

Bogons and Special Use Addresses

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

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

- 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

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

  Defined in RFC3330

  Examples: 127.0.0.0/8, 192.0.2.0/24

Ingress and Egress Route Filtering

- Two flavors of route filtering:
  - Distribute list—not so widely used, obsolete
    - ACL entries generate hit count
  - Prefix list—Primarily used
- Both work—engineering preference
- Two filtering techniques:
  - Explicit permit (permit then deny any)
  - Explicit deny (deny then permit any)

Prefix-List for a BGP Prefix List

```
ip prefix-list rfc1918-dsua seq 5 deny   0.0.0.0/8 le 32
ip prefix-list rfc1918-dsua seq 10 deny   10.0.0.0/8 le 32
ip prefix-list rfc1918-dsua seq 15 deny   127.0.0.0/8 le 32
ip prefix-list rfc1918-dsua seq 20 deny   169.254.0.0/16 le 32
ip prefix-list rfc1918-dsua seq 25 deny   172.16.0.0/12 le 32
ip prefix-list rfc1918-dsua seq 30 deny   192.0.2.0.0/24 le 32
ip prefix-list rfc1918-dsua seq 35 deny   192.168.0.0/16 le 32
ip prefix-list rfc1918-dsua seq 40 deny   224.0.0.0/3 le 32
ip prefix-list rfc1918-dsua seq 45 permit 0.0.0.0/0 le 32
```

BGP with Prefix-List Route Filtering

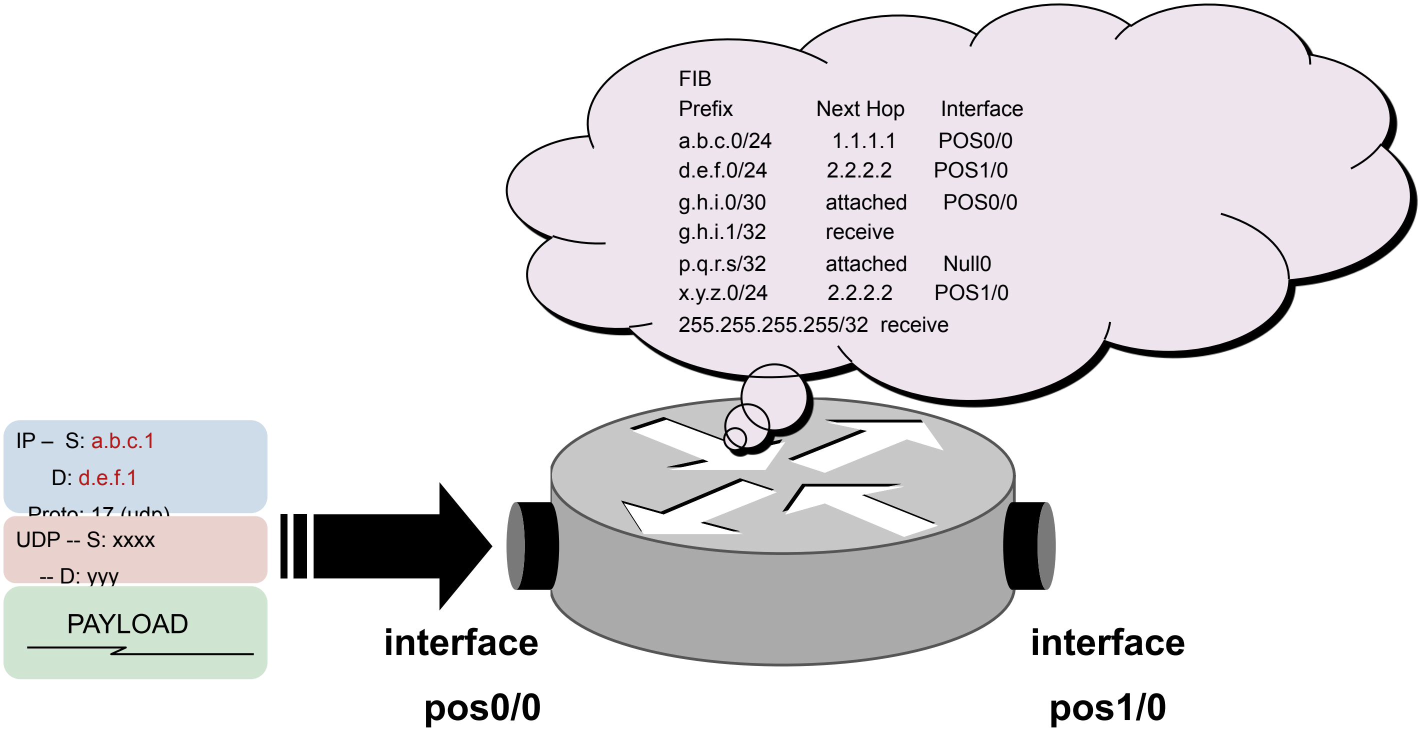
```
router bgp 65535
  no synchronization
  bgp dampening
  neighbor 220.220.4.1 remote-as 210
  neighbor 220.220.4.1 version 4
  neighbor 220.220.4.1 prefix-list rfc1918-dsua in
  neighbor 220.220.4.1 prefix-list rfc1918-dsua out
  neighbor 222.222.8.1 remote-as 220
  neighbor 222.222.8.1 version 4
  neighbor 222.222.8.1 prefix-list rfc1918-dsua in
  neighbor 222.222.8.1 prefix-list rfc1918-dsua out
  no auto-summary
!
```

Planes, Paths and Punts

The Four Planes

- Data plane—packets going through the router
- Control plane—routing protocols gluing the network together
- Management plane—tools and protocols used to manage the device
- Services plane—customer traffic (similar to the data plane), traffic requiring specialized forwarding functions applied it

It’s All About the Packet



FIB

Prefix	Next Hop	Interface
a.b.c.0/24	1.1.1.1	POS0/0
d.e.f.0/24	2.2.2.2	POS1/0
g.h.i.0/30	attached	POS0/0
g.h.i.1/32	receive	
p.q.r.s/32	attached	Null0
x.y.z.0/24	2.2.2.2	POS1/0
255.255.255.255/32	receive	

- Once a packet gets into the Internet, some device, somewhere has to do one of two things: [1] Forward the Packet* or [2] Drop the Packet
- In the context of security, the questions are more granular:

  Who forwarded the packet, and **what** resources were required to do so…

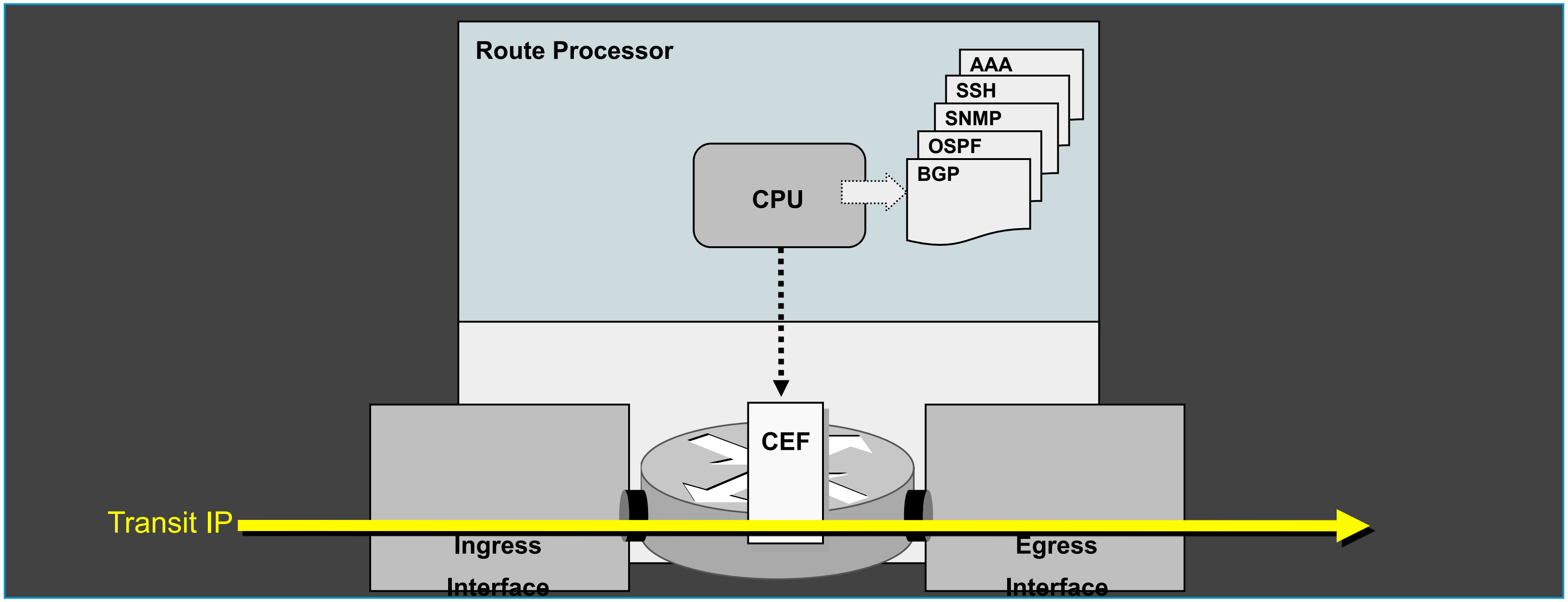
  Who dropped the packet, and **why** was it dropped…

* Forwarding Could Entail Adding a **Service** to the Packet as well…

Transit Packets

Transit Packets

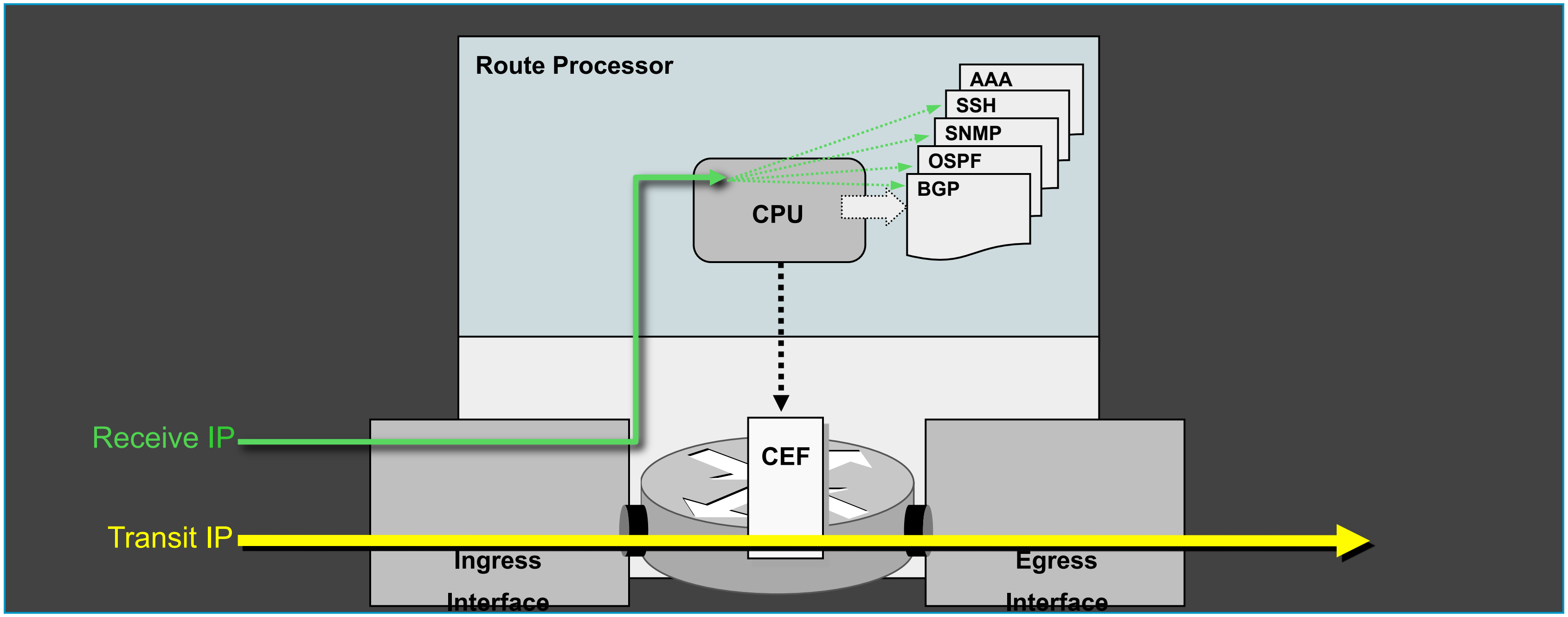
- Well-formed IP packets that follow standard, destination IP address-based forwarding processes. No extra processing by the route processor is required to forward these packets.
- The destination IP address of these packets is located downstream from the network device and thus, the packet is forwarded out an egress interface



Receive Packets

Receive Packets

- Packets that are destined to the network device itself (e.g. control and management packets) must be handled by the route processor CPU since they ultimately are destined for and handled by applications running at the process level.
- All of the packets in this set must be handled by the route processor



Receive Adjacencies

- CEF entries for traffic destined to router

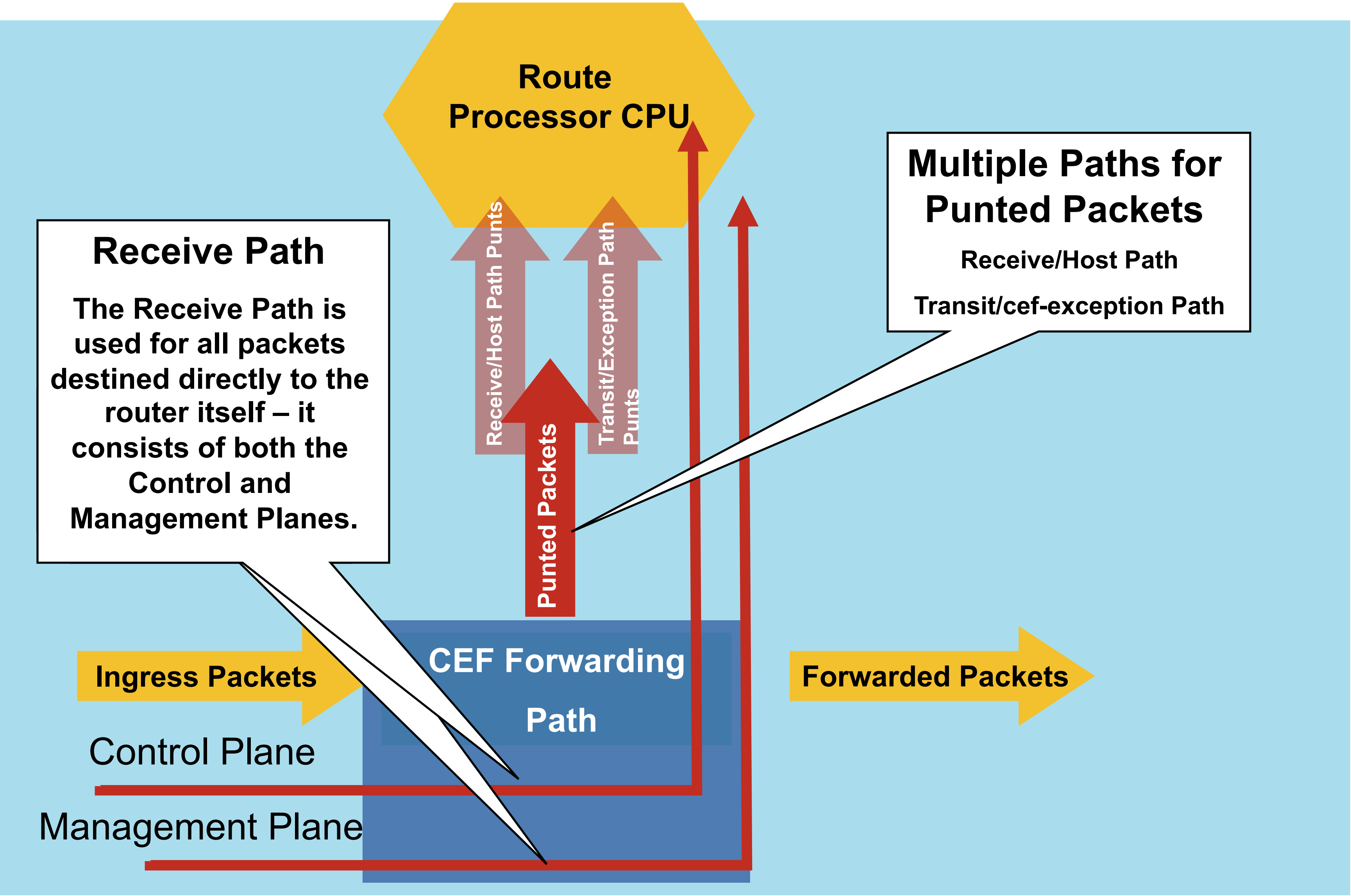
  Real interfaces and Loopbacks

  Broadcast and Multicast address space

```
router#sh ip cef
Prefix                     Next Hop          Interface
255.255.255.255/32          receive
10.1.2.0/24                172.16.1.216       GigabitEthernet3/0
10.1.3.0/24                172.16.1.216       GigabitEthernet3/0
224.0.0.0/24                receive
172.16.1.196/32             receive
```

- Packets with next hop receive are sent to the router for processing
- Traffic usually routing protocols, management, and multicast control traffic

Receive Path



Receive Path

The Receive Path is used for all packets destined directly to the router itself – it consists of both the Control and Management Planes.

Multiple Paths for Punted Packets

Receive/Host Path

Transit/cef-exception Path

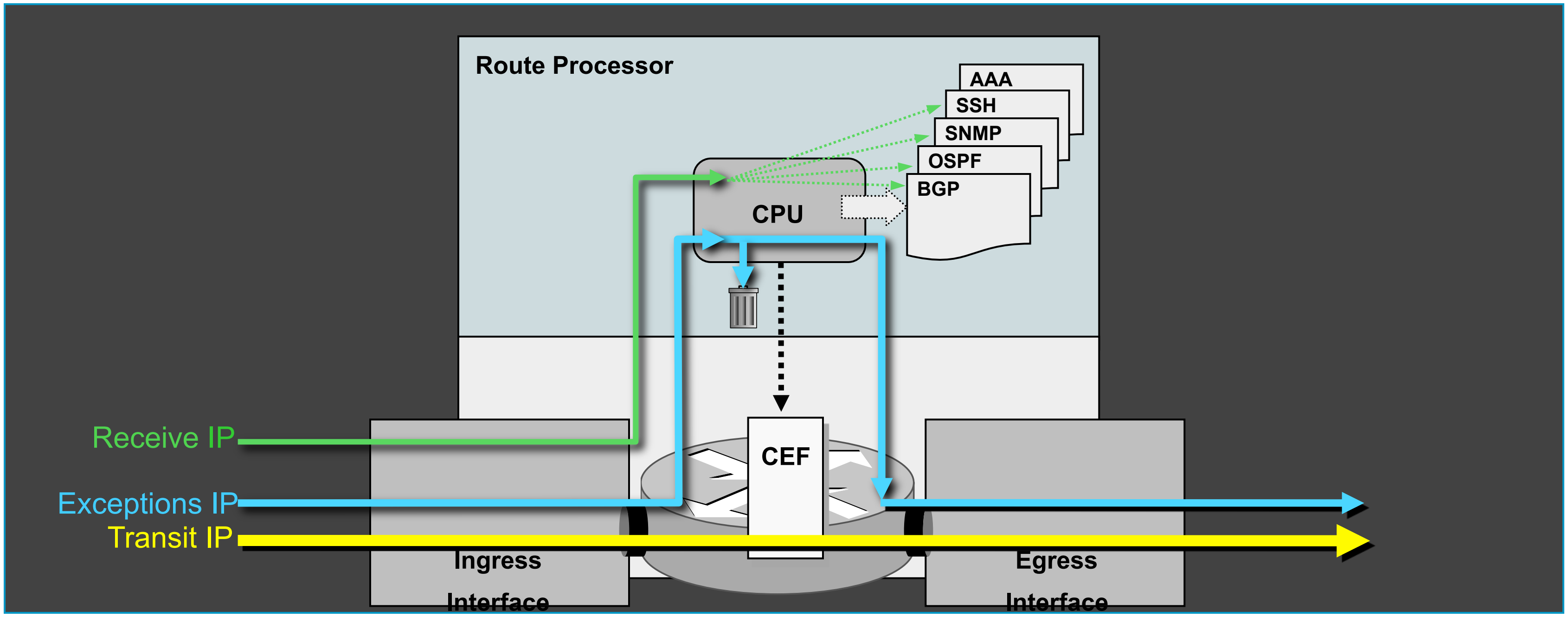
What Is a Punt?

- Receive adjacency
- Transit packets that need additional processing
    - Specific router configuration: ACL logging, Cisco IOS FW, etc.
    - IP Options set
    - Require fragmentation
    - ICMP Unreachables due to routing, MTU, or filtering
    - Expired TTL (ICMP Time Exceeded)
    - Destinations lacking a next-hop adjacency (ARP—CEF Glean punt)
    - Malformed fields (ICMP Parameter error)

Exceptions IP Packets

Exceptions IP Packets

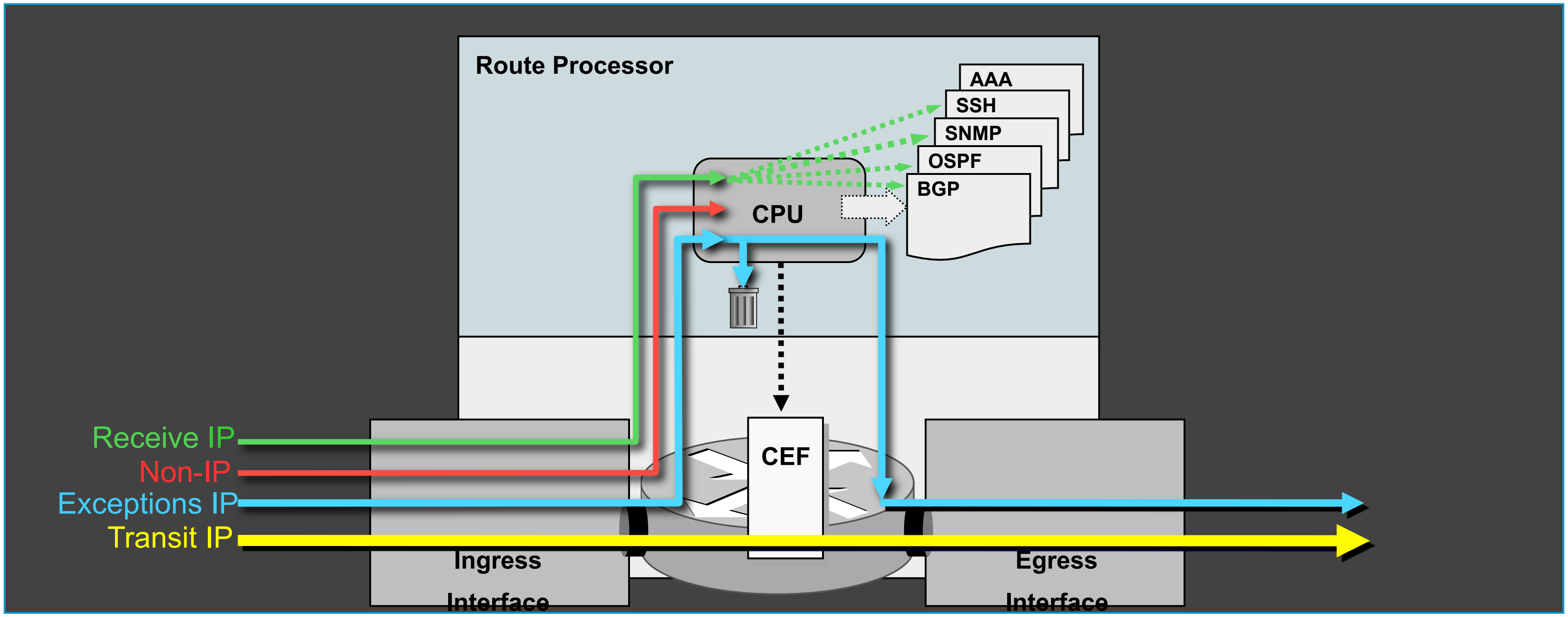
- Exception IP packets include, for example, IPv4 and IPv6 packets containing IP header options, IP packets with expiring TTLs, and certain transit IP packets under specific conditions, such as the first packet of a multicast flow or a new NAT session
- All of the packets in this set must be handled by the route processor



Non-IP Packets

Non-IP Packets

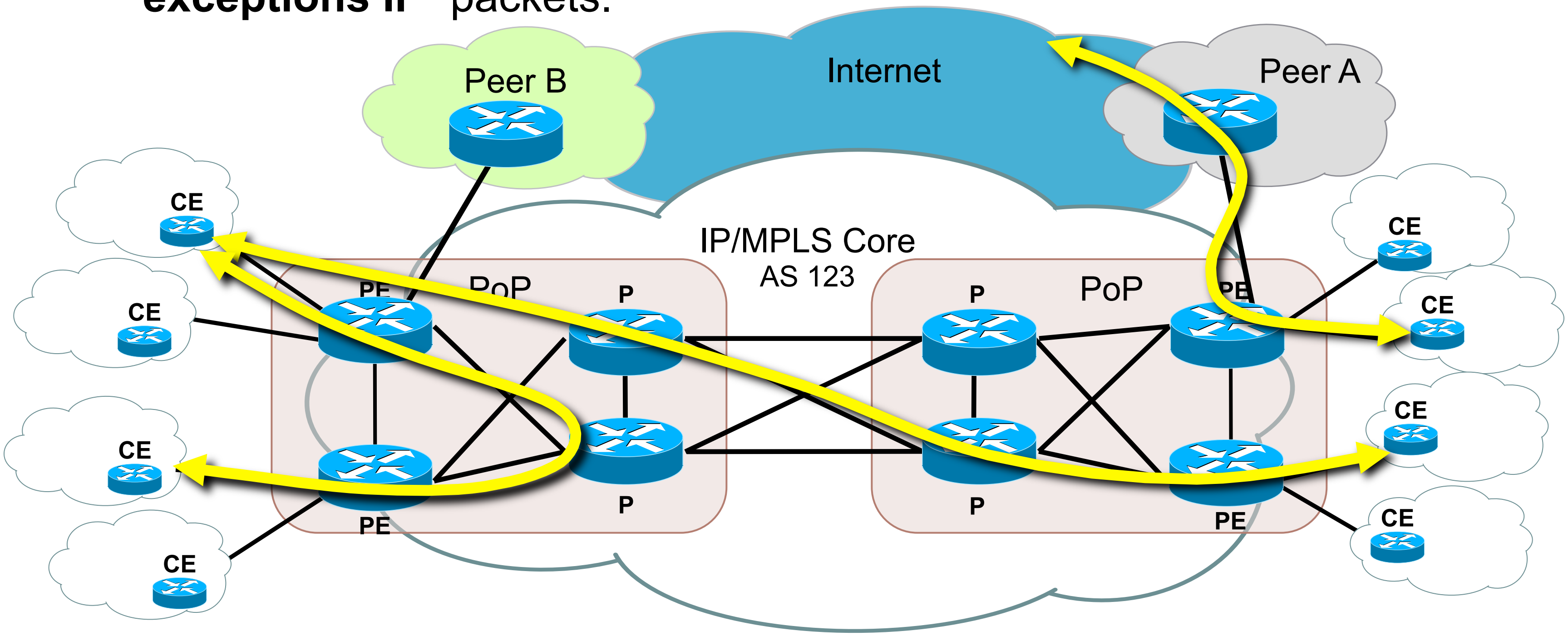
- Layer 2 keepalives, ISIS packets, Cisco Discovery Protocol (CDP) packets, and PPP Link Control Protocol (LCP) packets are examples of non-IP packets
- All of the packets in this set must be handled by the route processor



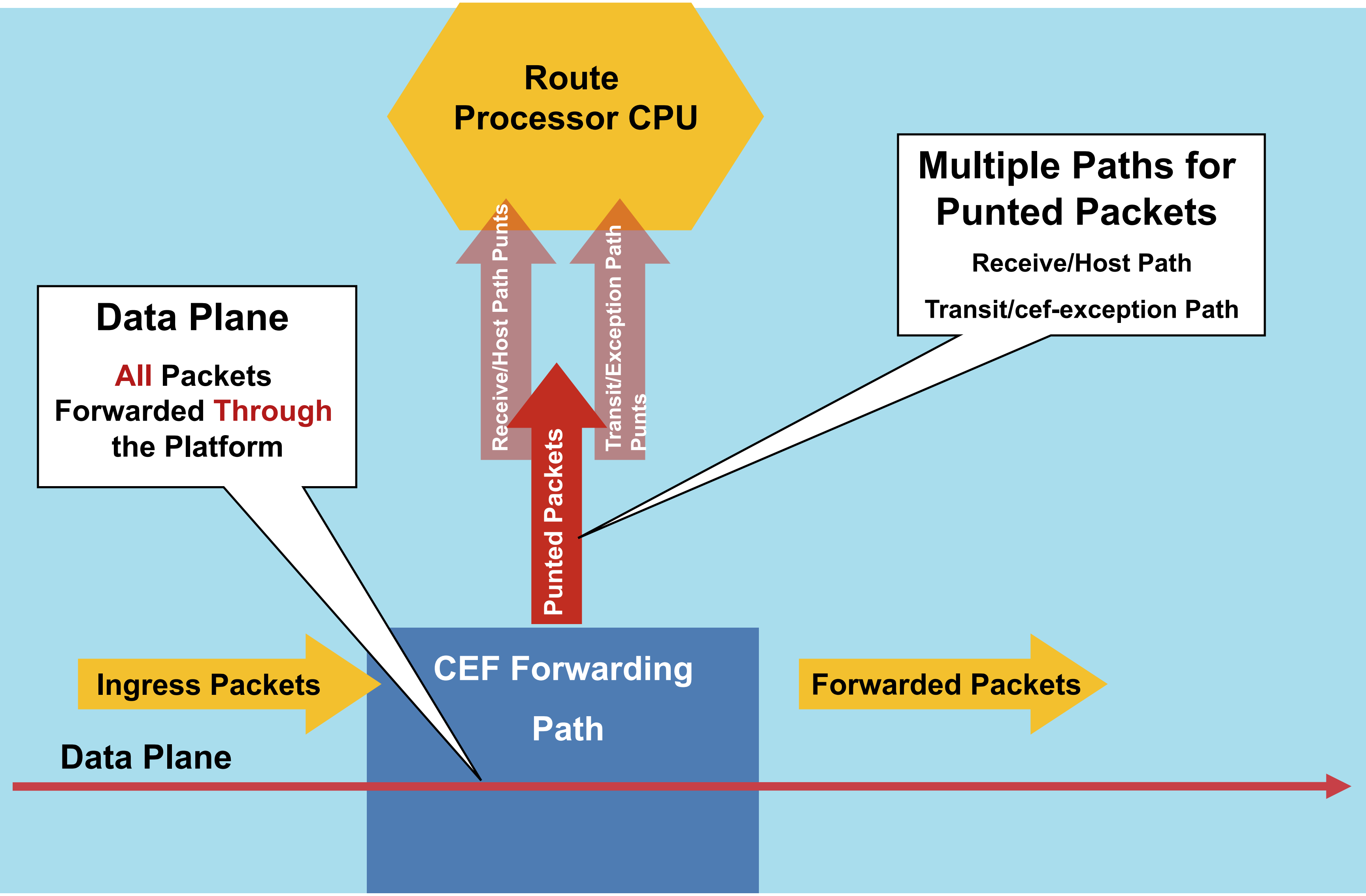
IP Data Plane

IP Data Plane

- The logical group containing all “**customer**” application traffic generated by hosts, clients, servers, and applications that are sourced from and destined to other devices
- Data plane traffic is always be seen as **transit** packets by network elements. Most will be forwarded in the fast path; some may be “**exceptions IP**” packets.



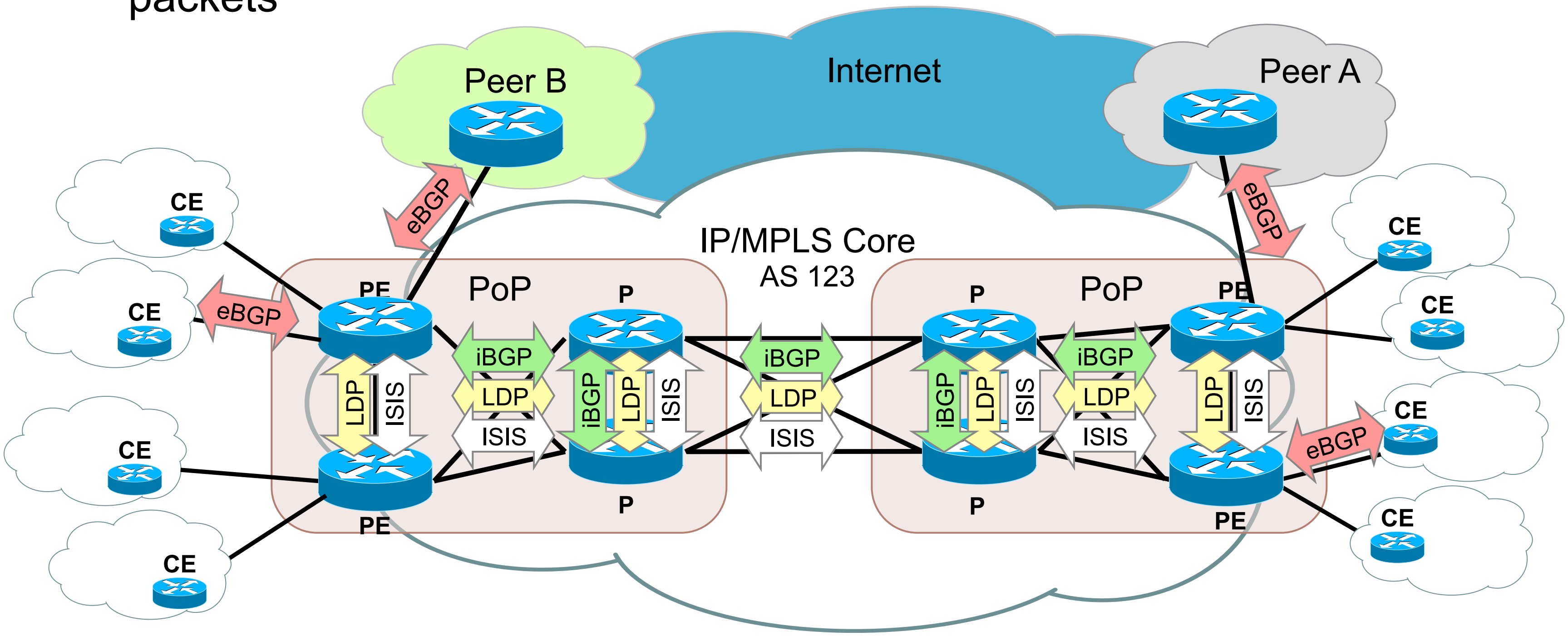
Data Plane


Route Processor CPU
Multiple Paths for Punted Packets
Receive/Host Path
Transit/cef-exception Path
Receive/Host Path Punts
Transit/Exception Path Punts
Punted Packets
Data Plane
All Packets Forwarded Through the Platform
CEF Forwarding Path
Ingress Packets
Forwarded Packets
Data Plane

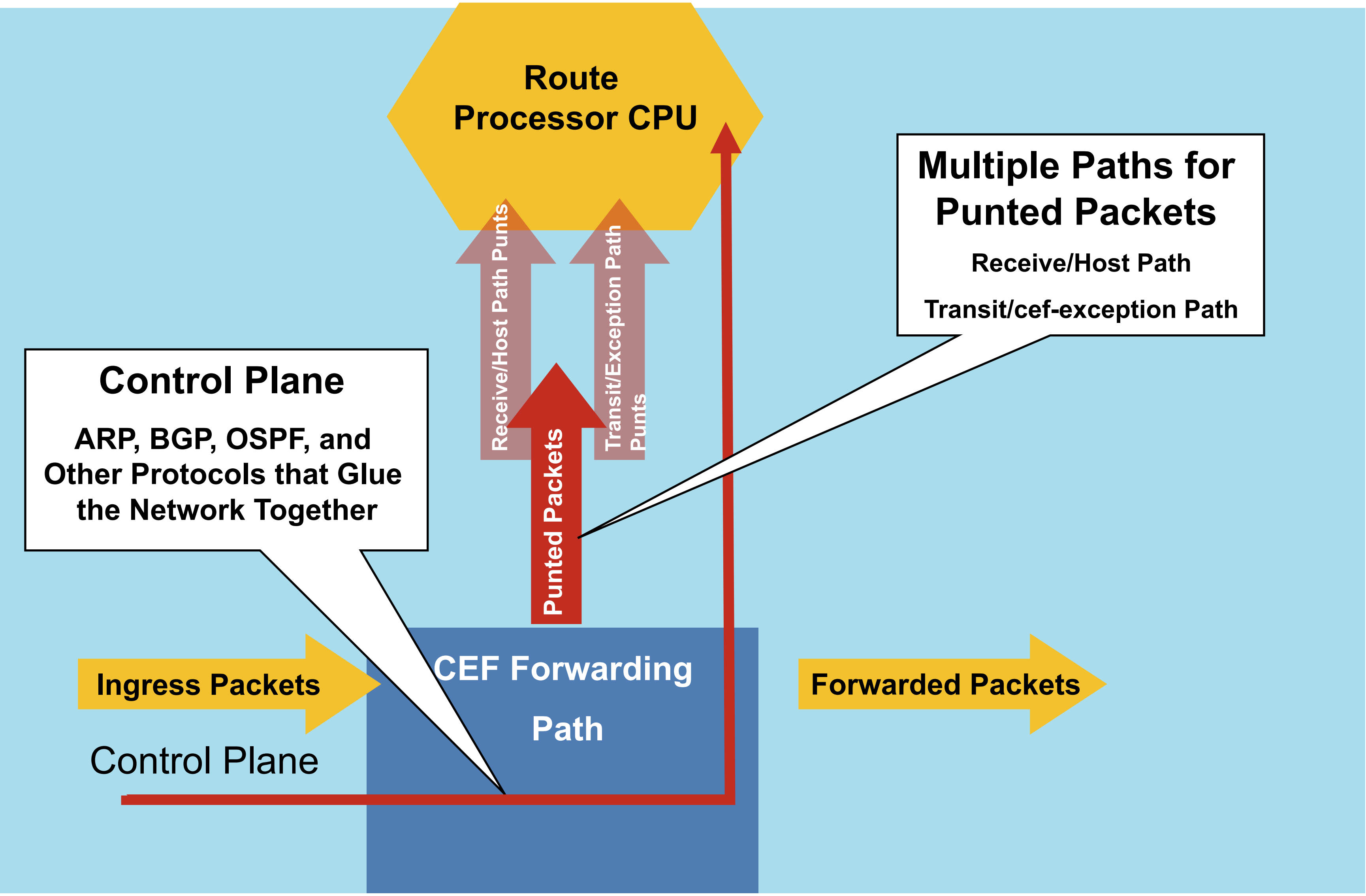
IP Control Plane

IP Control Plane

- The logical group containing all **routing**, **signaling**, **link-state**, and other control protocols used to create and maintain the state of the network and interfaces.
- Control plane traffic always includes **receive** packets from the perspective of the src/dst network element, but logically includes certain transit packets



Control Plane



Route Processor CPU

Multiple Paths for Punted Packets

Receive/Host Path

Transit/cef-exception Path

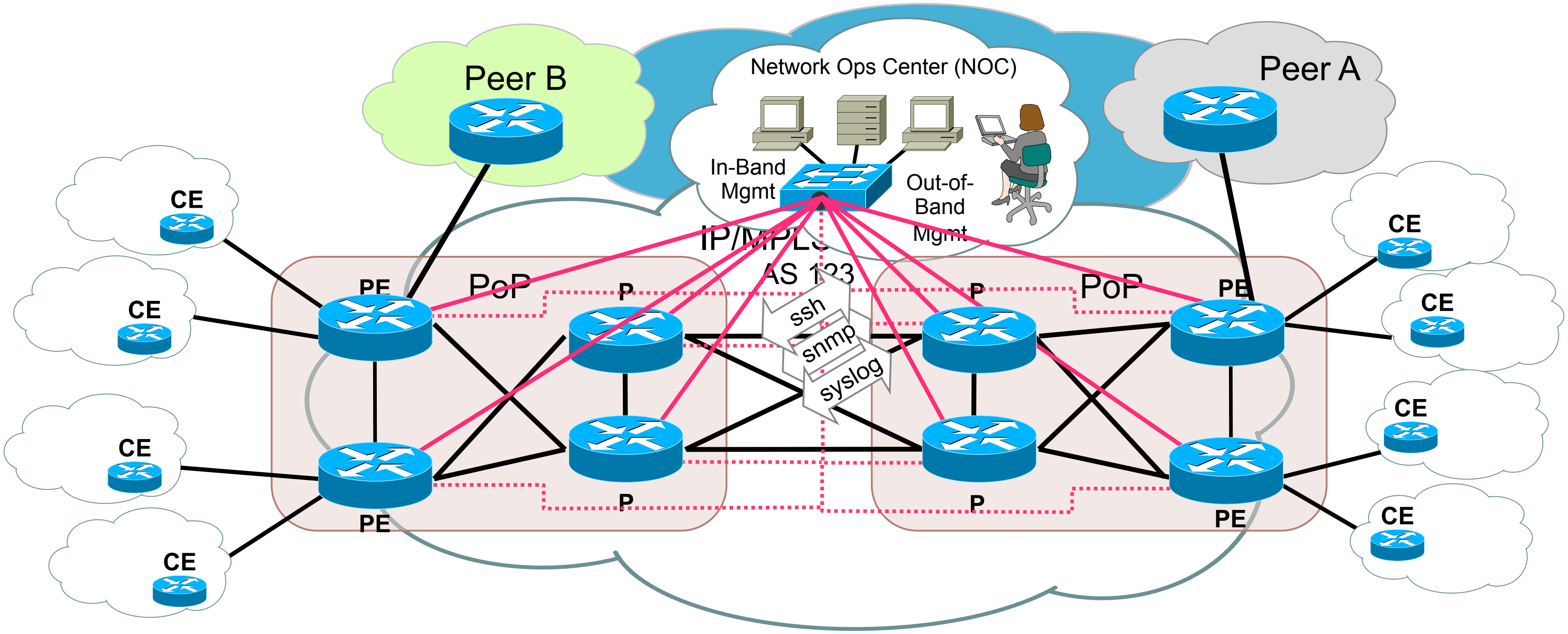
Control Plane

ARP, BGP, OSPF, and Other Protocols that Glue the Network Together

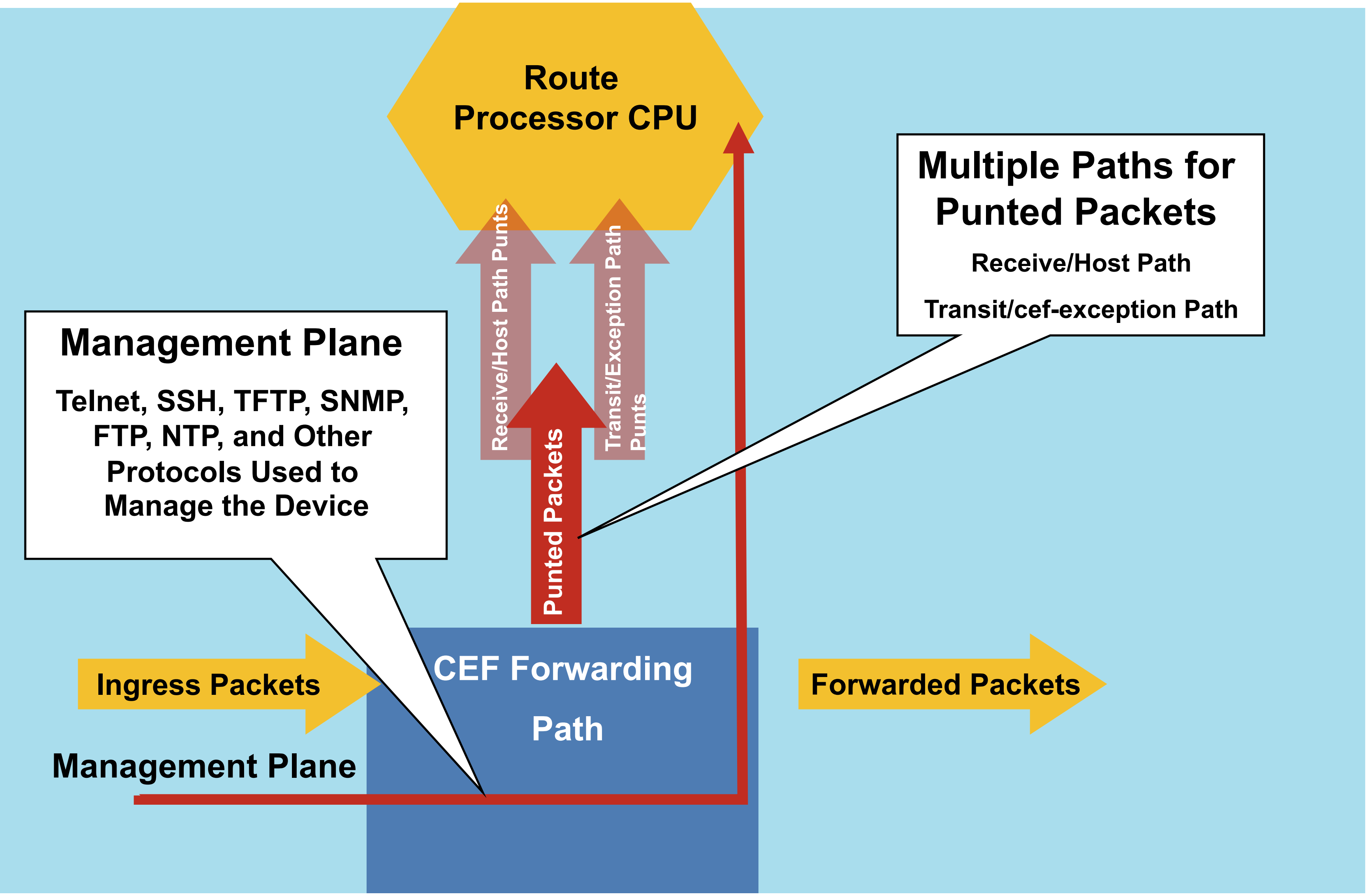
IP Management Plane

IP Management Plane

- The logical group containing all **management** traffic supporting provisioning, maintenance, and monitoring functions for the network..
- Management plane traffic always includes **receive** packets from the perspective of the src/dst network element, but logically includes certain transit packets



Management Plane



Multiple Paths for Punted Packets

Receive/Host Path

Transit/cef-exception Path

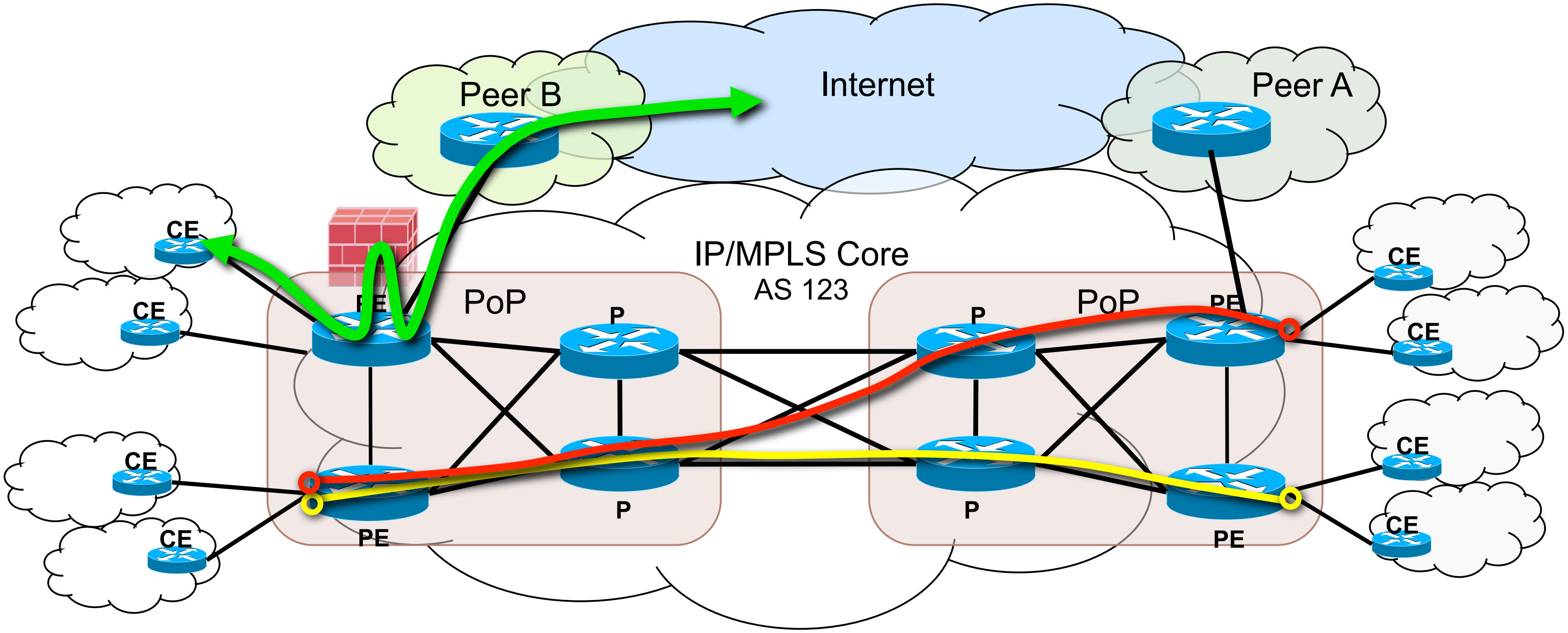
Management Plane

Telnet, SSH, TFTP, SNMP, FTP, NTP, and Other Protocols Used to Manage the Device

IP Services Plane

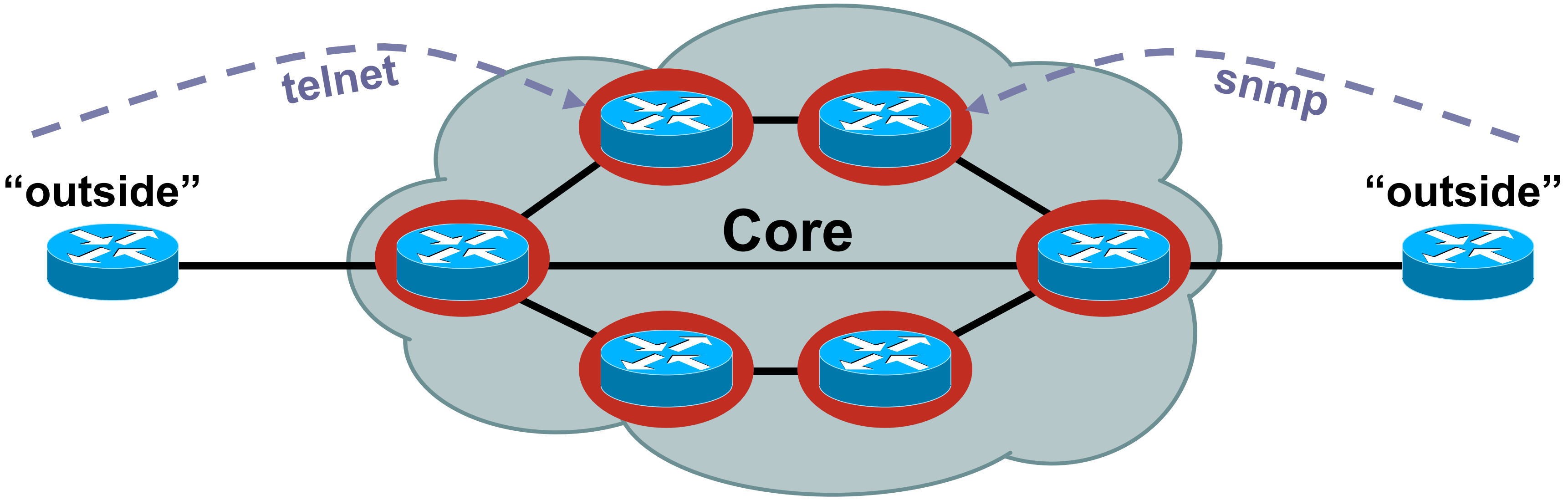
IP Services Plane

- The logical group containing “**customer**” traffic (like the data plane), but with the major difference that this traffic requires specialized forwarding functions applied it, and possibly consistent handling applied end to end.
- Services plane traffic is “**transit**” traffic, but network elements use special handling to apply or enforce the intended policies for various service types



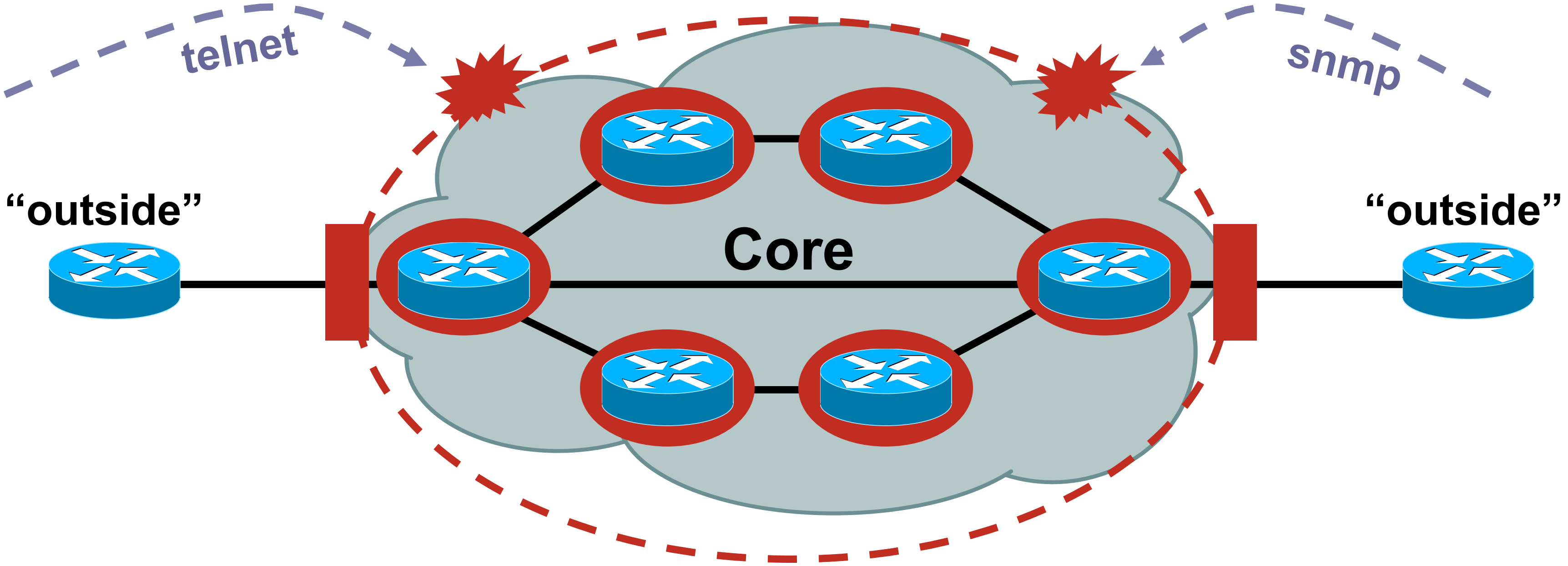
Infrastructure Security

The Old World



- Core routers individually secured
- Every router accessible from outside

The New World



- Core routers individually secured plus
- Infrastructure protection
- Routers generally not accessible from outside

RFC 2827/BCP 38

RFC 2827/BCP 38 Ingress Packet Filtering

- Packets should be sourced from valid, allocated address space, consistent with the topology and space allocation

Internet Connectivity Guidelines for BCP38

- Networks connecting to the Internet

  Must use inbound and outbound packet filters to protect the network

- Configuration example

  Outbound—only allow my network source addresses out

  Inbound—only allow specific ports to specific destinations in

BCP 38: Consequences of No Action

No BCP 38 Means That:

- Devices can (wittingly or unwittingly) send traffic with spoofed and/or randomly changing source addresses out to the network
- Complicates traceback immensely
- Sending bogus traffic is not free

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
- Dynamic ACLs with AAA profiles
- Unicast RPF strict mode
- IP source guard
- Cable source verify (DHCP)

Using ACLs to Enforce BCP38

- Static ACLs are the traditional method of ensuring that source addresses are not spoofed:

  Permit all traffic whose source address equals the allocation block

  Deny any other packet

- Principles:

  Filter as close to the edge as possible

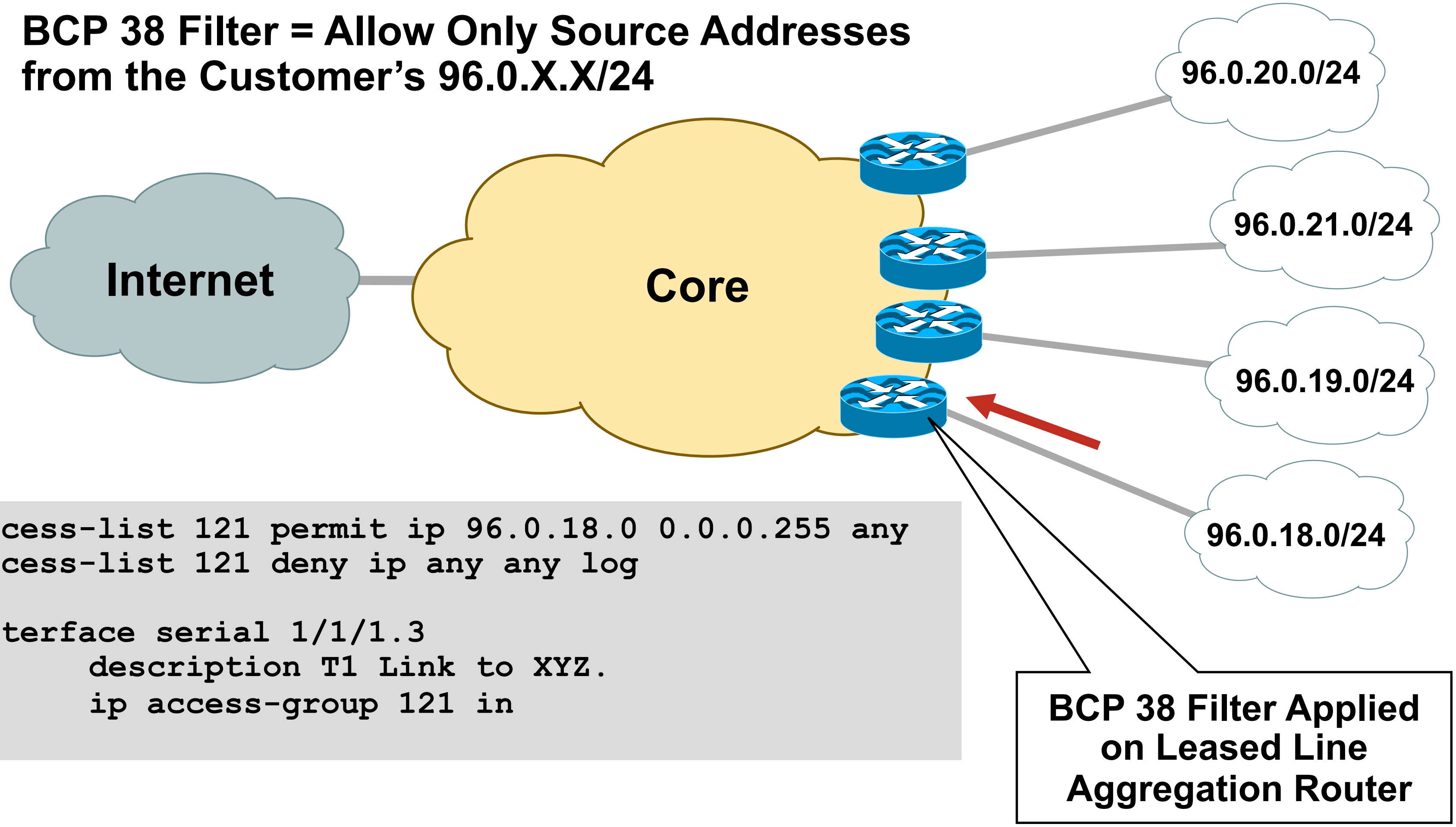
  Filter as precisely as possible

  Filter both source and destination where possible

Static ACLs for BCP 38 Ingress Packet Filtering

Allocation Block: 96.0.0.0/19

BCP 38 Filter = Allow Only Source Addresses from the Customer's 96.0.X.X/24

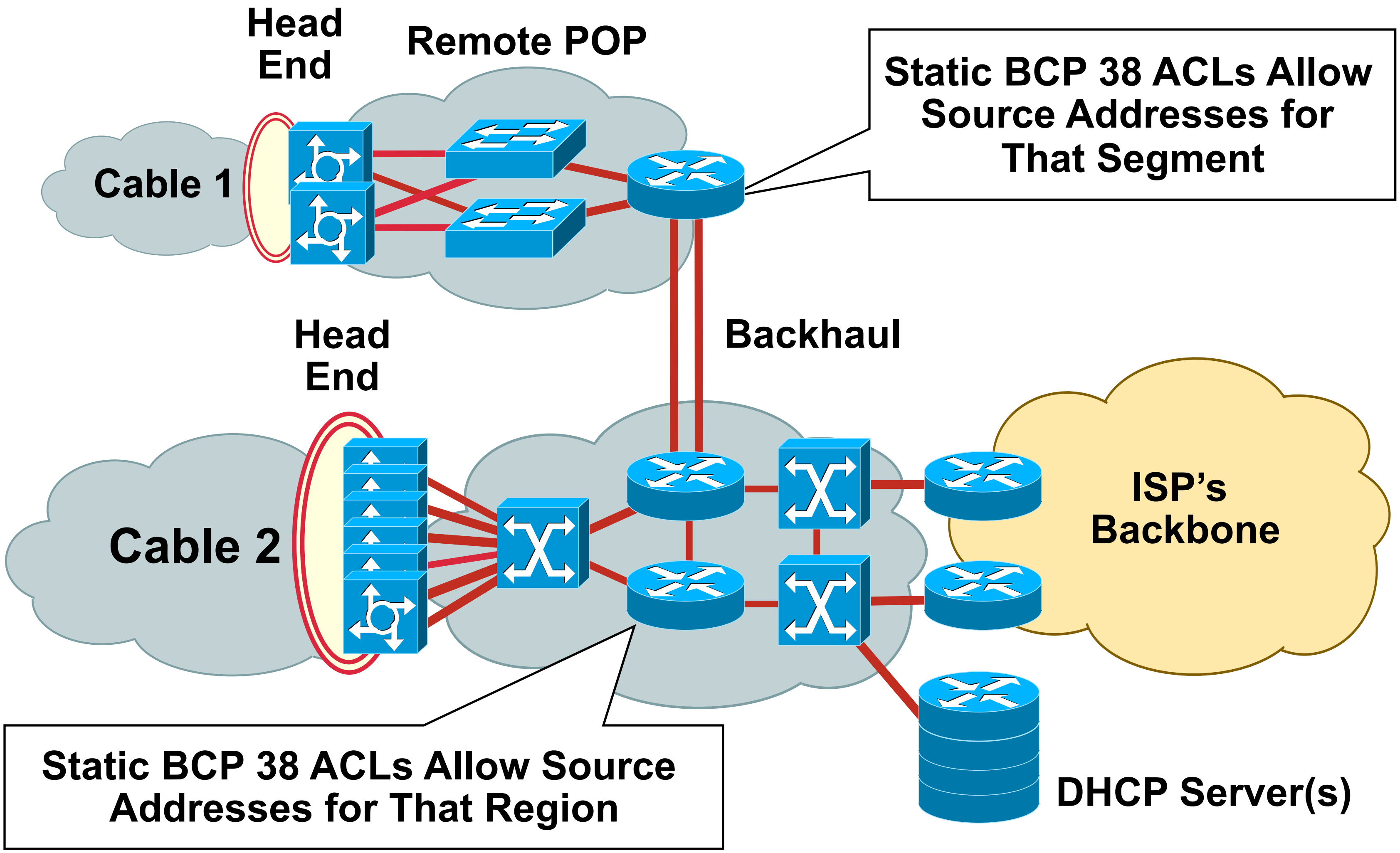


```
access-list 121 permit ip 96.0.18.0 0.0.0.255 any
access-list 121 deny ip any any log
!
interface serial 1/1/1.3
      description T1 Link to XYZ.
      ip access-group 121 in
!
```

BCP 38 Filter Applied on Leased Line Aggregation Router

ISP

Static BCP 38 ACLs: DHCP



Static BCP 38 ACLs Allow Source Addresses for That Segment

Static BCP 38 ACLs Allow Source Addresses for That Region

BCP ACL Guidelines

- ISPs

  Make sure your customers install filters on their routers - give them a template they can use

- Customer end-sites

  Make sure you install strong filters on routers you use to connect to the Internet

  First line of defense - never assume your ISP will do it

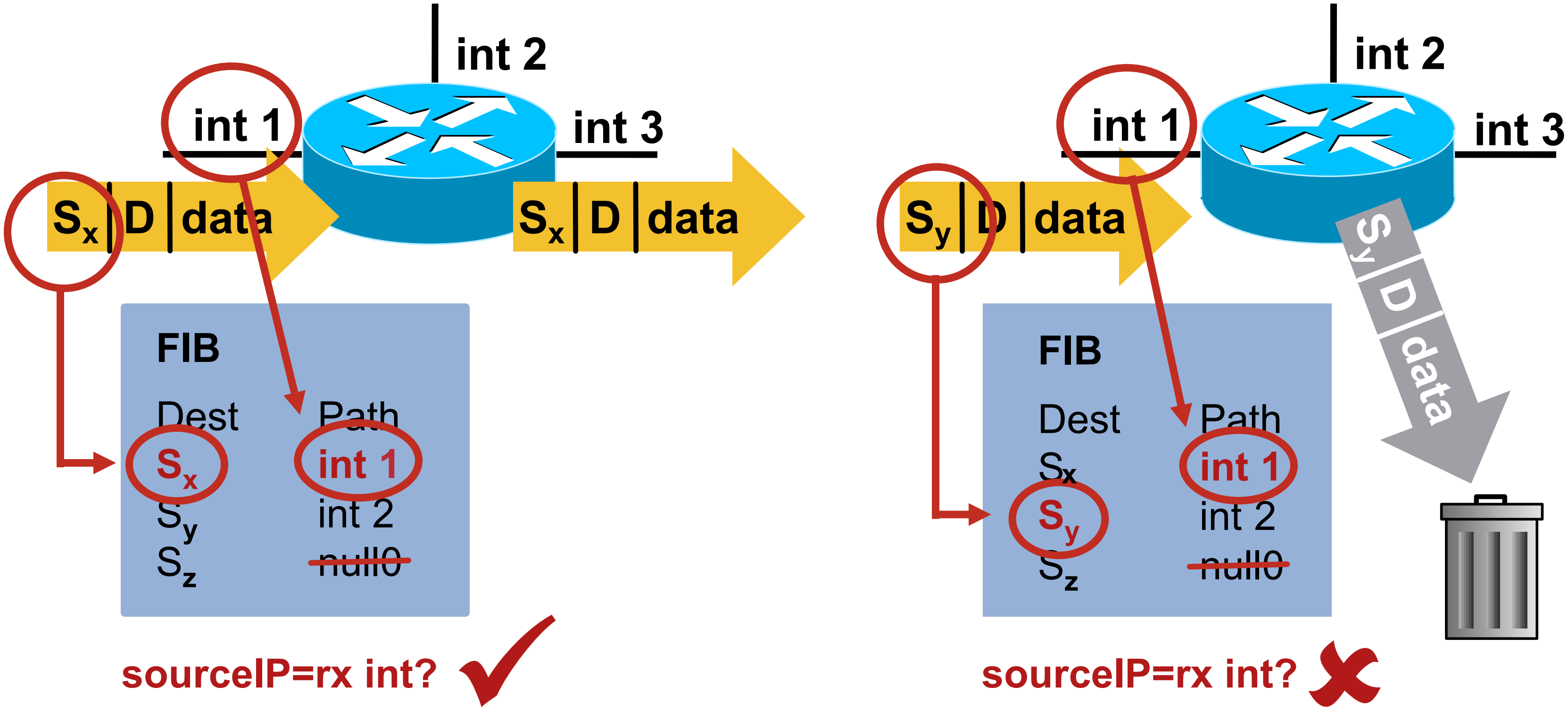
Unicast Reverse Path Forwarding (uRPF)

- CEF is required
- The purported source of ingress IP packets is checked to ensure that the route back to the source is “valid”
- Two flavors of uRPF:
  - Strict mode uRPF
  - Loose mode uRPF

uRPF—Strict Mode

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

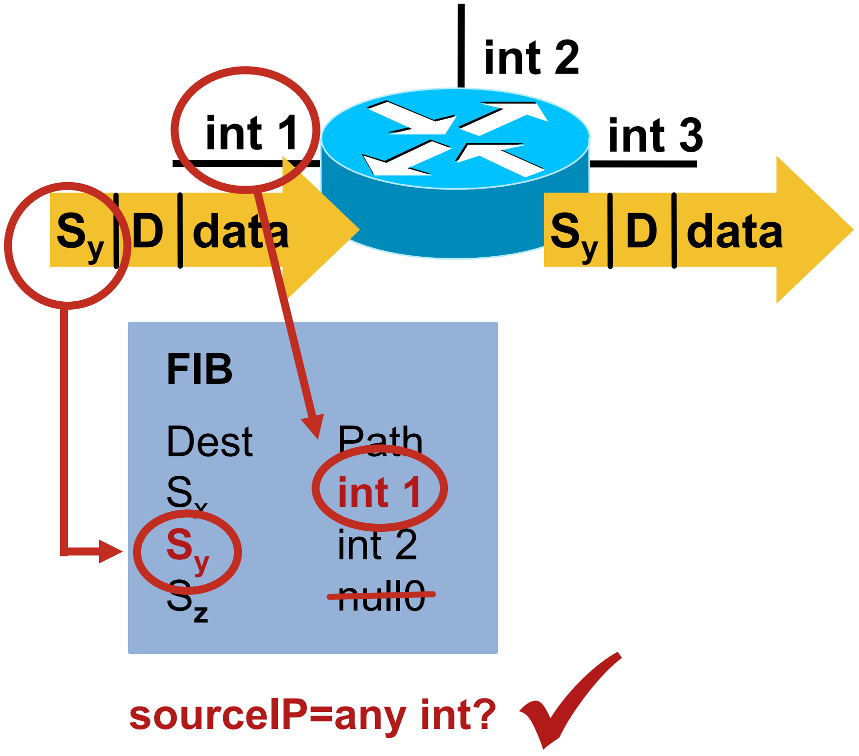
(deprecated syntax: ip verify unicast reverse-path)

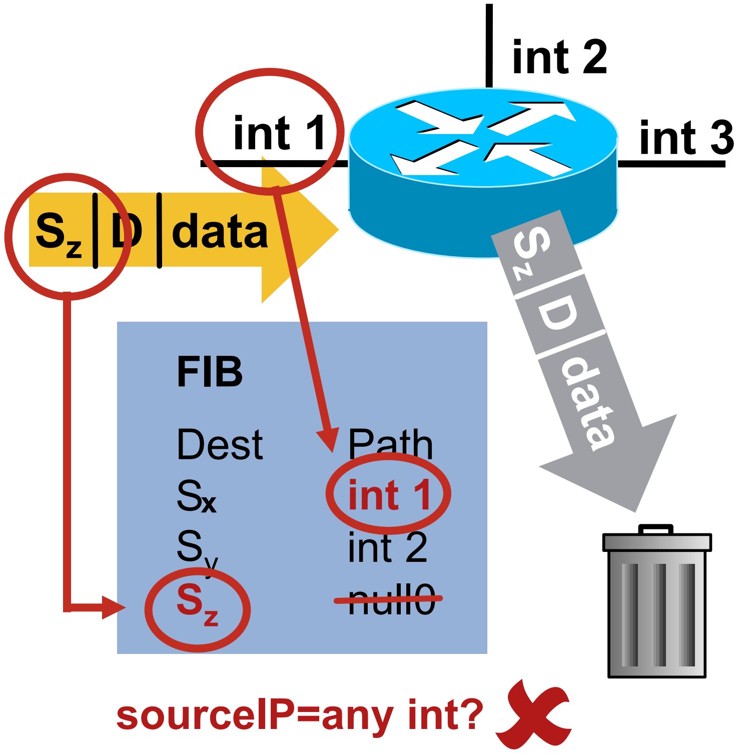


IP Verify Unicast Source Reachable—Via rx

uRPF—Loose Mode

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



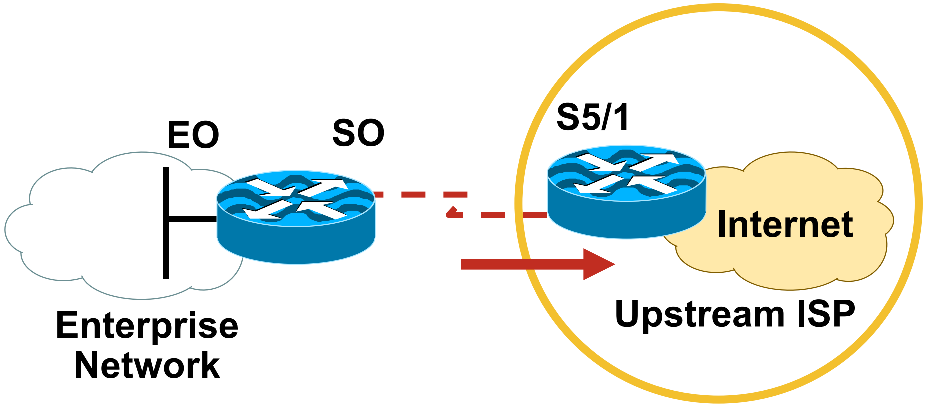


IP Verify Unicast Source Reachable—Via any

Unicast RPF (Strict Mode)

Simple Single Homed Customer Example

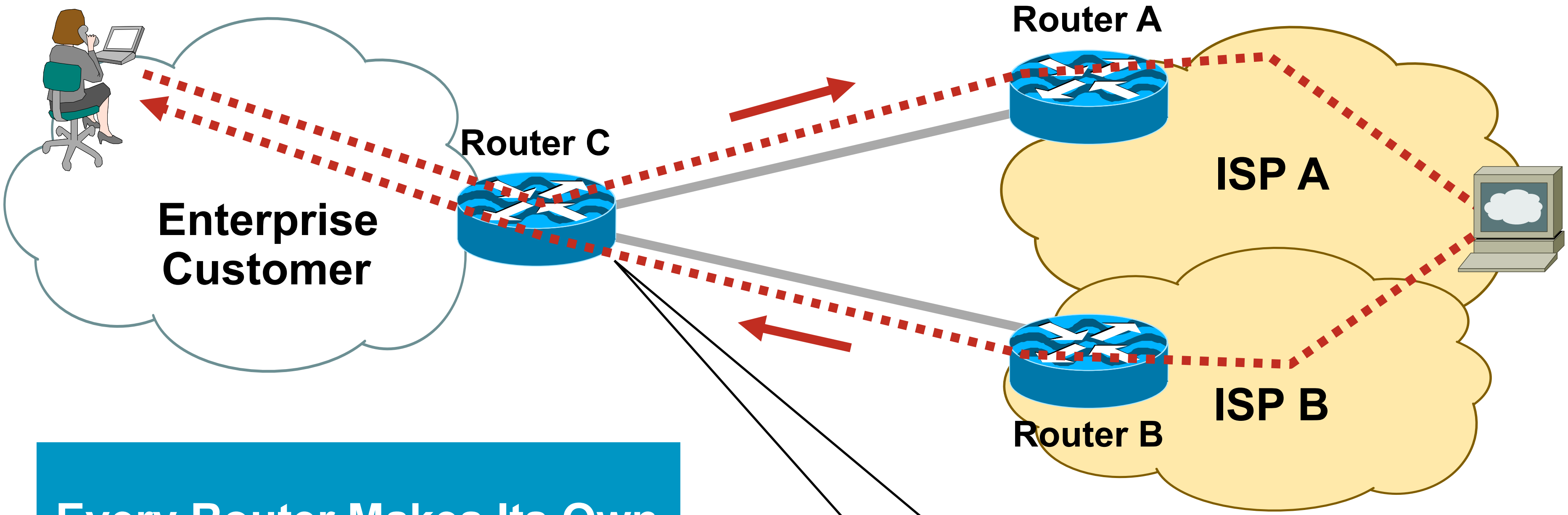
ISP Using uRPF for Ingress Filtering



```
interface Serial 5/1
 description 128K HDLC link to Galaxy Publications Ltd [galpub1]
 bandwidth 128
 ip unnumbered loopback 0
 !Unicast RPF activated
 ip verify unicast source reachable-via rx
 no ip redirects
 no ip directed-broadcast
 no ip proxy-arp
```

uRPF and Multihomed Customers

What Is Asymmetrical Routing?



Every Router Makes Its Own Best Path Forwarding Decision—Resulting in Asymmetrical Routing

Strict uRPF on This i/f Will Drop Traffic from the Server

Strict uRPF and Asymmetric Routing

- Traffic originating from multihomed customers can be verified with uRPF
- Solution: make routing symmetric
- Details in ISP Essentials:

  ftp://ftp-eng.cisco.com/cons/isp/security
  (a must-read for all SP engineers)

- Loose vs. Strict uRPF reference:

  Unicast Reverse Path Forwarding Loose Mode

  http://www.cisco.com/en/US/products/sw/iosswrel/ps1839/products_feature_guide09186a00803fa70b.html

Static BCP 38 Filtering in DHCP Networks

- Many broadband cable and DSL networks use DHCP for their CPE client provisioning
- DHCP works per shared segment, hence BCP 38 filters can be applied on the gateway router(s)

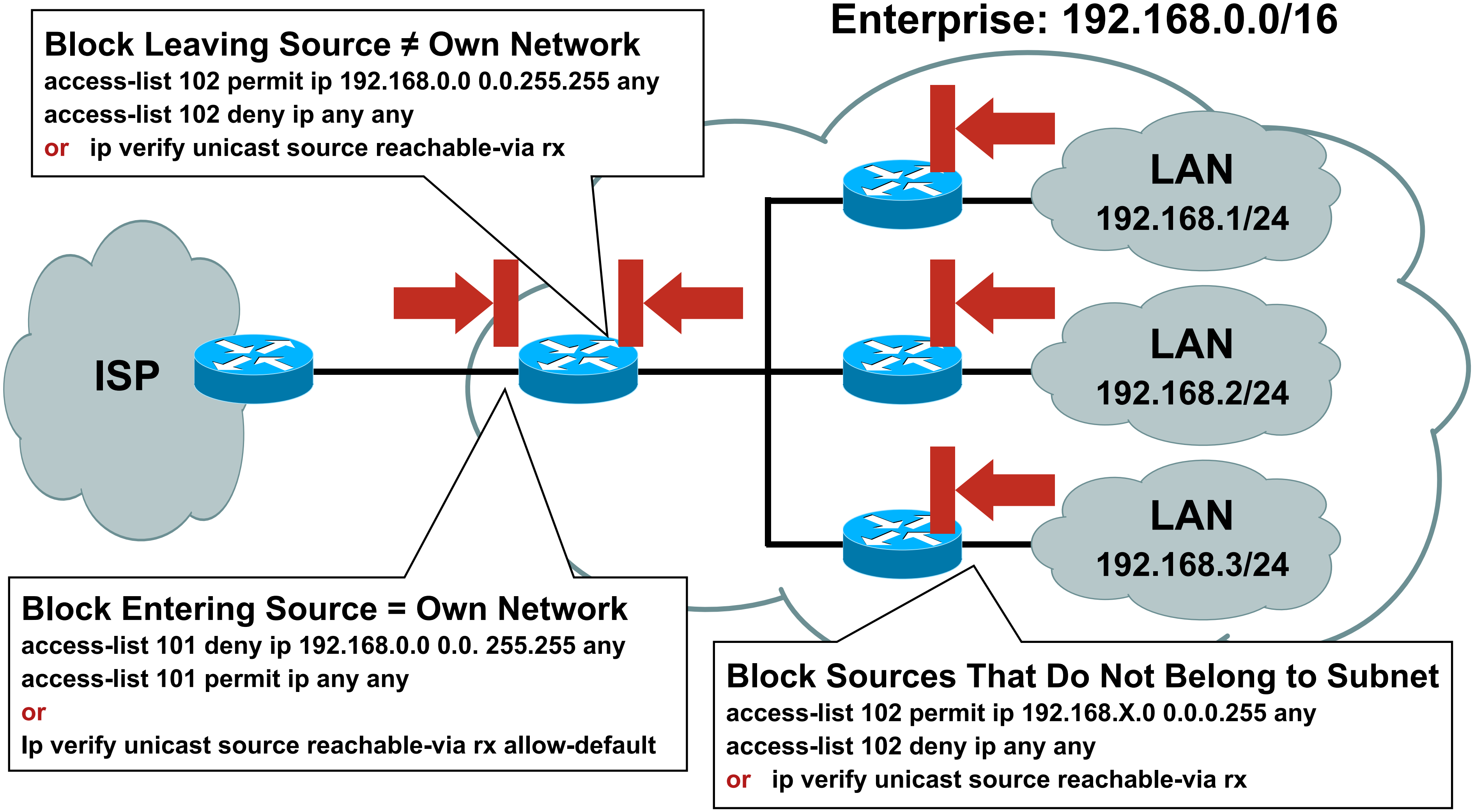
  Limitation is that people on the same segment can spoof each other

- For Ethernet-based networks we have IP source guard on Cisco Catalysts

  IP source verification that is also DHCP aware

  http://www.cisco.com/en/US/products/hw/switches/ps4324/products_configuration_guide_chapter09186a008019d0c8.html

Address Spoofing Prevention in the Enterprise



Block Leaving Source ≠ Own Network

access-list 102 permit ip 192.168.0.0 0.0.255.255 any
access-list 102 deny ip any any
or ip verify unicast source reachable-via rx

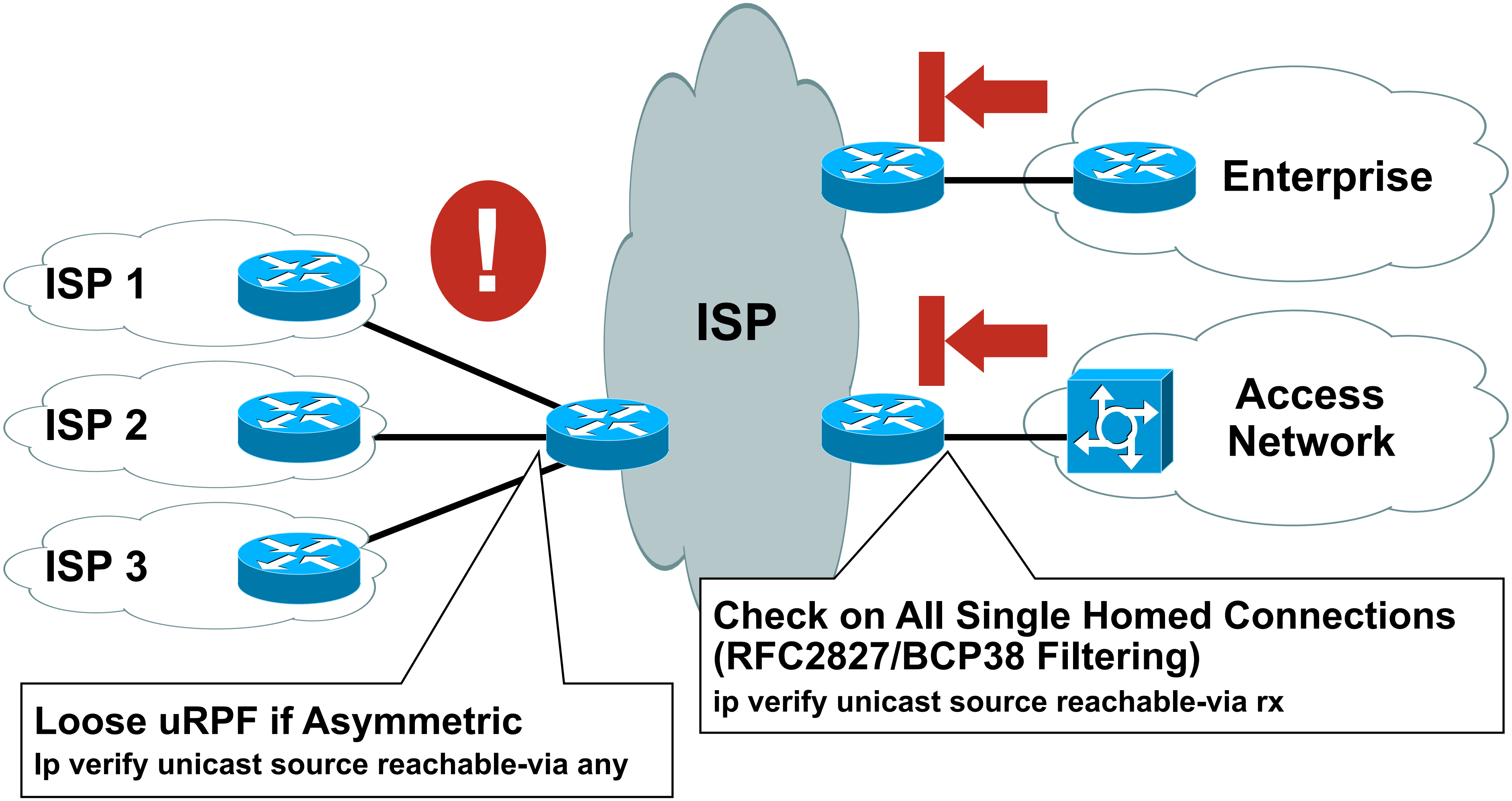
Block Entering Source = Own Network

access-list 101 deny ip 192.168.0.0 0.0. 255.255 any
access-list 101 permit ip any any
or
Ip verify unicast source reachable-via rx allow-default

Block Sources That Do Not Belong to Subnet

access-list 102 permit ip 192.168.X.0 0.0.0.255 any
access-list 102 deny ip any any
or ip verify unicast source reachable-via rx

Address Spoofing Prevention on the SP Network



Loose uRPF if Asymmetric

Ip verify unicast source reachable-via any

Check on All Single Homed Connections (RFC2827/BCP38 Filtering)

ip verify unicast source reachable-via rx

BCP 38 Filtering: Summary

- BCP 38 is an operational reality

  It works, it is scalable

  It is operationally deployable and maintainable

  It works on a wide variety of equipment

  Deployable in the vast majority of situations—
  no more excuses

- Take time to understand source address validation techniques, see which ones will work for you

- Find ways to gain operational confidence in the BCP 38 techniques

- BCP 84 lists specific filtering methods

Network Telemetry

SNMP, RMON and Their Ilk

Types of Network Telemetry

- SNMP
- NetFlow
- RMON
- BGP
- Syslog
- Packet capture
- Others

SNMP

- SNMP = Simple Network Management Protocol
- Canonical method of obtaining real-time information from network devices
- SNMPv3 provides authentication, encryption
- MIBs support polling of statistics ranging from interface bandwidth to CPU utilization to chassis temperature, etc.
- Both a “pull” model for statistical polling and a “push” model for trap generation based upon events such as link up/down
- Many open-source and commercial collection systems, visualization tools
- Easiest way to get into profiling of general network characteristics

SNMP: Net-Snmp Toolset

- Formerly known as UCD-SNMP toolset
- Open source SNMP command-line tools, library, trap-generator, agent, etc. available from

  http://www.net-snmp.org/
- Included with most Linux distros, FreeBSD, etc.
- Command-line access to SNMP data from enabled routers, switches, etc.
- Runs on Linux, FreeBSD, Mac OS/X, Solaris, other *NIX, Windows
- Perl modules available via CPAN

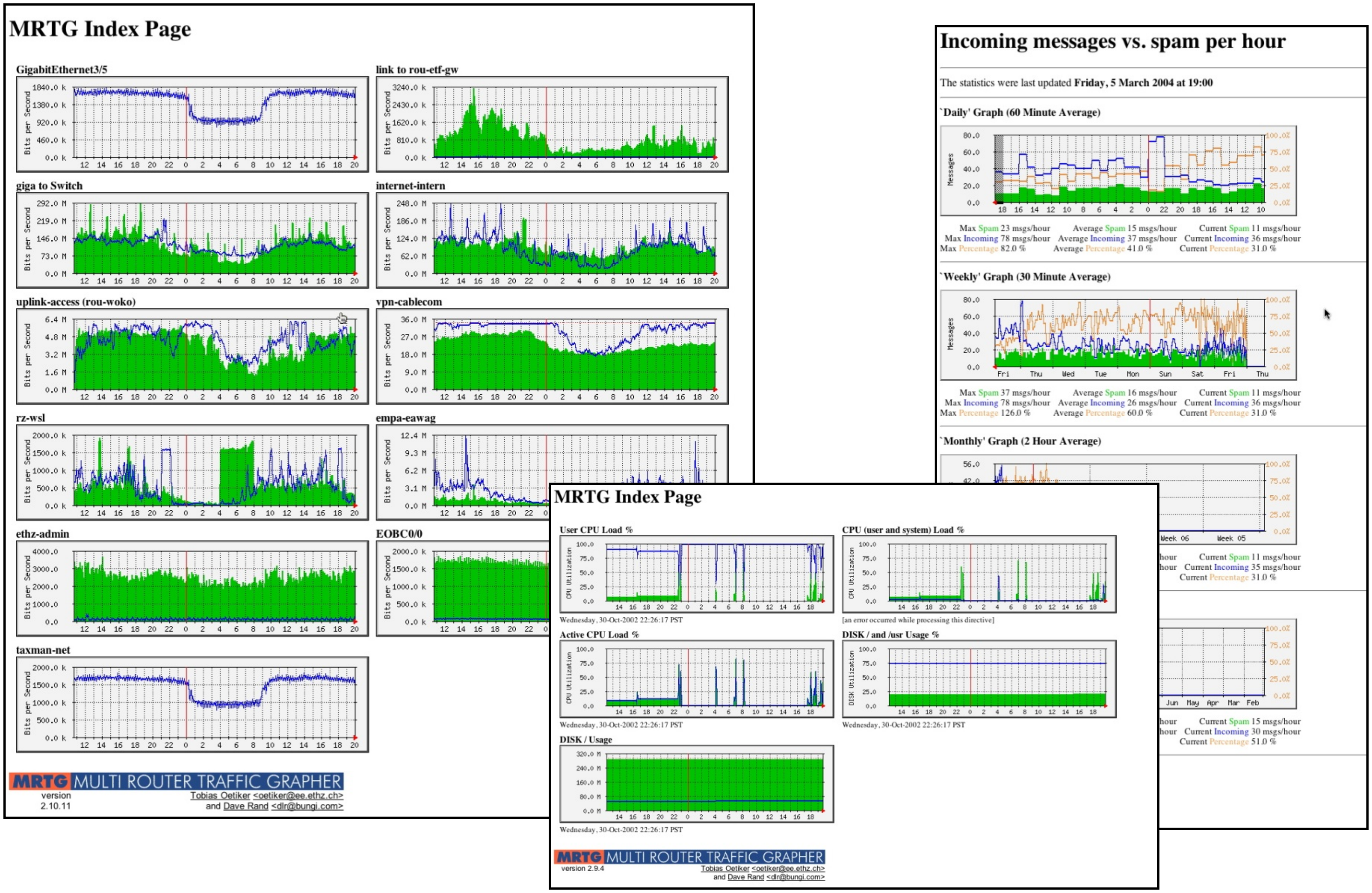
SNMP: MRTG

- MRTG—the Multi Router Traffic Grapher
- Open source SNMP visualization toolset developed by Tobi Oetiker, available from

  http://oss.oetiker.ch/mrtg/

- Long track-record—(in general use since 1995)
- Can be used to graph router/switch data, host performance information from systems running SNMP agents, etc. (generates HTML w/PNG images)
- Runs on Linux, FreeBSD, Mac OS/X, Solaris, other *NIX, Windows
- Written in Perl, has its own SNMP implementation

Example: MRTG Graphs



MRTG Index Page

GigabitEthernet3/5

link to rou-etf-gw

giga to Switch

internet-intern

uplink-access (rou-woko)

vpn-cablecom

rz-wsl

empa-eawag

ethz-admin

EOBC0/0

taxman-net

MRTG MULTI ROUTER TRAFFIC GRAPHER
version 2.10.11
Tobias Oetiker <oetiker@ee.ethz.ch>
and Dave Rand <dlr@bungi.com>

Incoming messages vs. spam per hour

The statistics were last updated **Friday, 5 March 2004 at 19:00**

`Daily' Graph (60 Minute Average)

Max Spam 23 msgs/hour Average Spam 15 msgs/hour Current Spam 11 msgs/hour
Max Incoming 78 msgs/hour Average Incoming 37 msgs/hour Current Incoming 36 msgs/hour
Max Percentage 82.0 % Average Percentage 41.0 % Current Percentage 31.0 %

`Weekly' Graph (30 Minute Average)

Max Spam 37 msgs/hour Average Spam 16 msgs/hour Current Spam 11 msgs/hour
Max Incoming 78 msgs/hour Average Incoming 26 msgs/hour Current Incoming 36 msgs/hour
Max Percentage 126.0 % Average Percentage 60.0 % Current Percentage 31.0 %

`Monthly' Graph (2 Hour Average)

hour Current Spam 11 msgs/hour
hour Current Incoming 35 msgs/hour
Current Percentage 31.0 %

hour Current Spam 15 msgs/hour
hour Current Incoming 30 msgs/hour
Current Percentage 51.0 %

MRTG Index Page

User CPU Load %

Wednesday, 30-Oct-2002 22:26:17 PST

CPU (user and system) Load %

[an error occurred while processing this directive]

Active CPU Load %

Wednesday, 30-Oct-2002 22:26:17 PST

DISK / and /usr Usage %

Wednesday, 30-Oct-2002 22:26:17 PST

DISK / Usage

Wednesday, 30-Oct-2002 22:26:17 PST

MRTG MULTI ROUTER TRAFFIC GRAPHER
version 2.9.4
Tobias Oetiker <oetiker@ee.ethz.ch>
and Dave Rand <dlr@bungi.com>

Source: mrtg.org

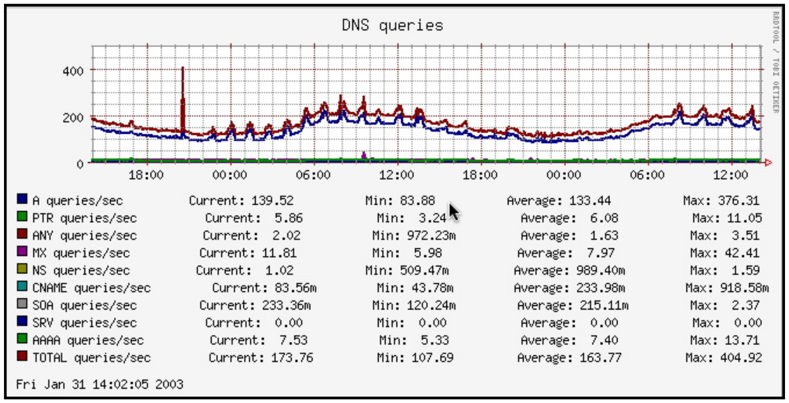
SNMP: RRDTool

- RRDTool—the Round Robin Database Tool
- Another open source SNMP visualization toolset developed by Tobi Oetiker, available from

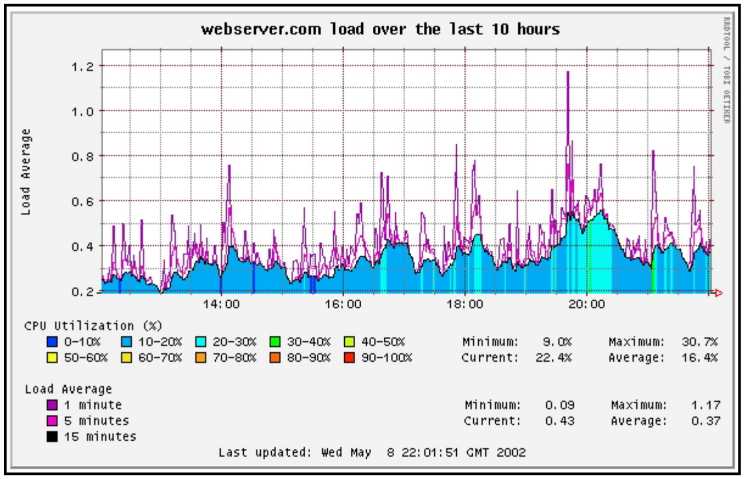
  http://oss.oetiker.ch/rrdtool/

- Improved graphing performance, new types of graphs
- Can be used in conjunction with MRTG—does not do its own SNMP collection (can also be used w/NetFlow via OSU flow-tools and FlowScan)
- Runs on Linux, FreeBSD, Mac OS/X, Solaris, other *NIX, Windows
- Many nice HTML/PHP front-ends such as Cacti, Cricket, Big Sister, etc.

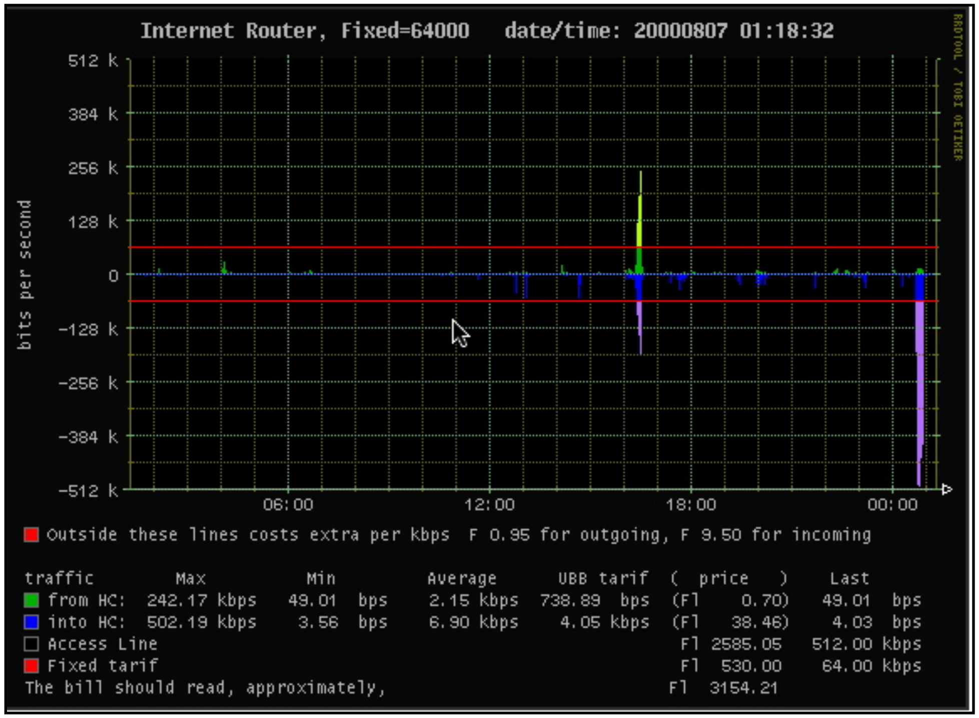
Example: RRDTool Graphs



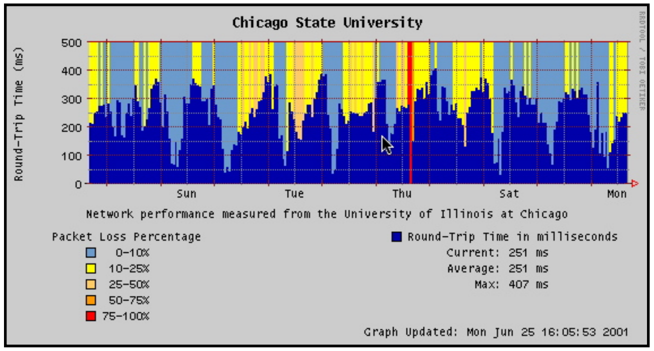
```
■ A queries/sec          Current: 139.52      Min: 83.88       Average: 133.44      Max: 376.31
■ PTR queries/sec         Current:  5.86       Min:  3.24       Average:  6.08       Max: 11.05
■ ANY queries/sec         Current:  2.02       Min: 972.23m     Average:  1.63       Max:  3.51
■ MX queries/sec         Current: 11.81        Min:  5.98       Average:  7.97       Max: 42.41
■ NS queries/sec         Current:  1.02       Min: 509.47m     Average: 989.40m     Max:  1.59
■ CNAME queries/sec        Current: 83.56m      Min: 43.78m      Average: 233.98m     Max: 918.58m
■ SOA queries/sec        Current: 233.36m      Min: 120.24m     Average: 215.11m     Max:  2.37
■ SRV queries/sec         Current:  0.00       Min:  0.00       Average:  0.00       Max:  0.00
■ AAAA queries/sec         Current:  7.53       Min:  5.33       Average:  7.40       Max: 13.71
■ TOTAL queries/sec        Current: 173.76      Min: 107.69      Average: 163.77      Max: 404.92

Fri Jan 31 14:02:05 2003
```



```
CPU Utilization (%)
    ■ 0-10%   ■ 10-20%  ■ 20-30%  ■ 30-40%  ■ 40-50%      Minimum:   9.0%     Maximum:  30.7%
    ■ 50-60%  ■ 60-70%  ■ 70-80%  ■ 80-90%  ■ 90-100%     Current:  22.4%     Average:  16.4%

Load Average
    ■ 1 minute                                              Minimum:   0.09     Maximum:   1.17
    ■ 5 minutes                                             Current:   0.43     Average:   0.37
    ■ 15 minutes
                  Last updated: Wed May  8 22:01:51 GMT 2002
```



```
■ Outside these lines costs extra per kbps  F 0.95 for outgoing, F 9.50 for incoming

traffic        Max          Min          Average      UBB tarif  (  price   )    Last
■ from HC:  242.17 kbps   49.01  bps    2.15 kbps  738.89  bps  (Fl     0.70)   49.01  bps
■ into HC:  502.19 kbps    3.56  bps    6.90 kbps    4.05 kbps  (Fl    38.46)    4.03  bps
□ Access Line                                                    Fl  2585.05    512.00 kbps
■ Fixed tarif                                                    Fl   530.00     64.00 kbps
The bill should read, approximately,                            Fl  3154.21
```



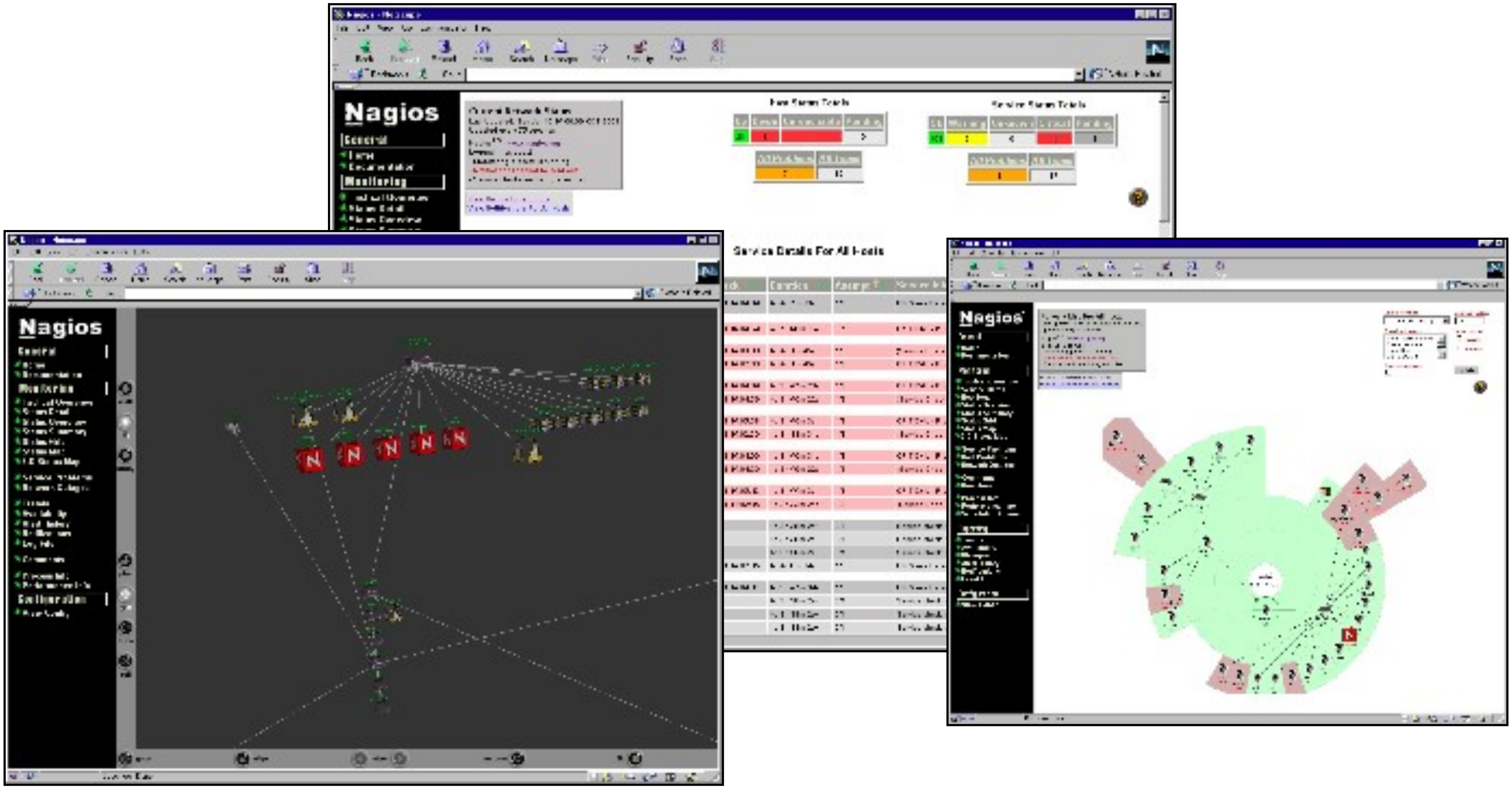
```
      Network performance measured from the University of Illinois at Chicago

Packet Loss Percentage                           ■ Round-Trip Time in milliseconds
    ■   0-10%                                          Current: 251 ms
    ■  10-25%                                          Average: 251 ms
    ■  25-50%                                              Max: 407 ms
    ■  50-75%
    ■ 75-100%
                                            Graph Updated: Mon Jun 25 16:05:53 2001
```

Source: http://people.ee.ethz.ch/~oetiker/webtools/rrdtool/

SNMP: NMS

- Network Management Systems (NMS) can serve as SNMP consoles, among other things
- Many can use SNMP traps and/or other forms of telemetry as triggers for paging, scripted actions, etc.
- Pulling information together can be useful for NOCs, operations teams
- Commercial systems such as HP OpenView, Micromuse NetCool, IBM Tivoli, CA Unicenter
- Several open source systems—Big Brother (http://bb4.com/), Big Sister (http://bigsister.graeff.com/), Nagios (http://www.nagios.org/), and others

Nagios Examples

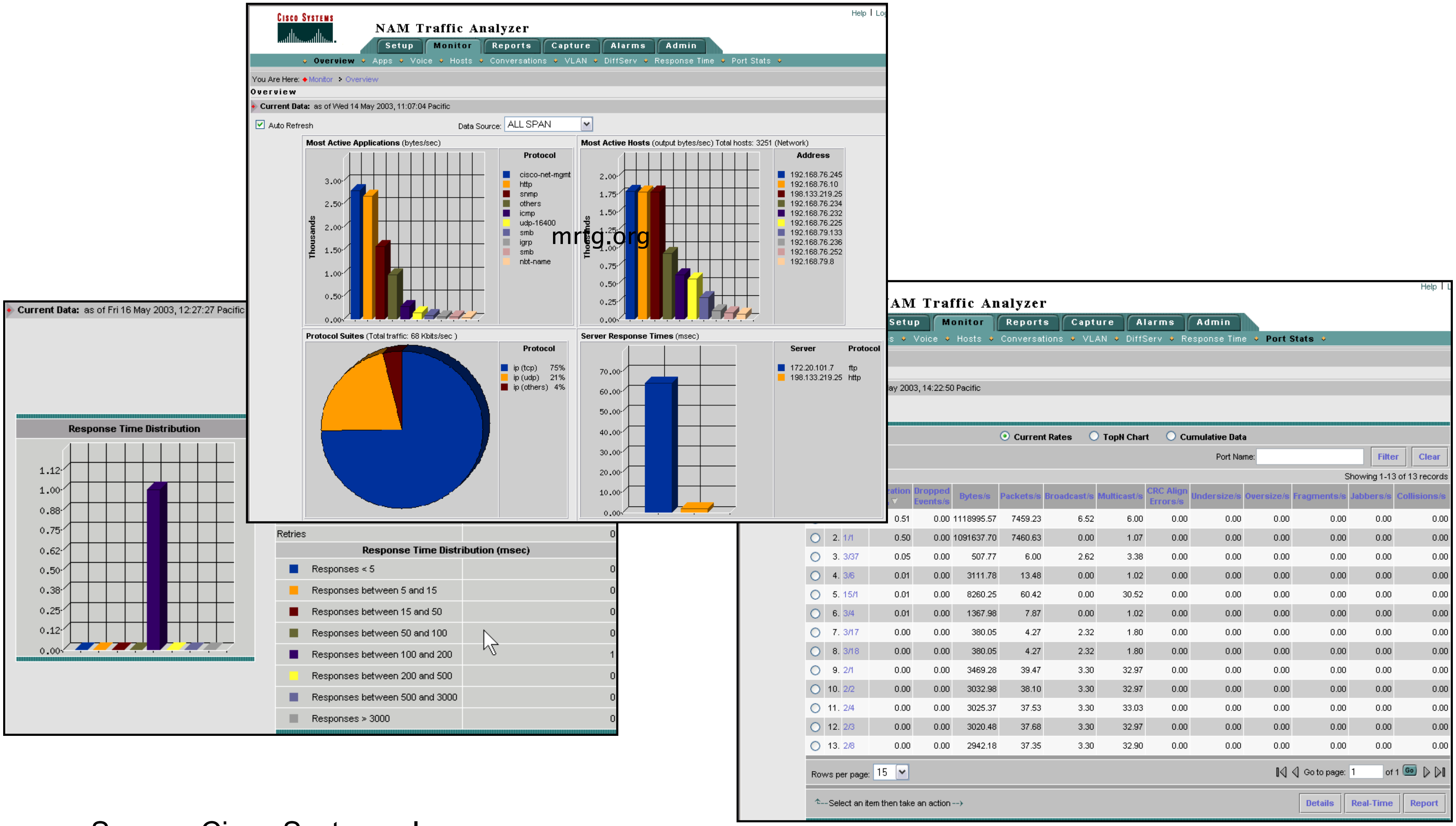


Source: http://www.nagios.org

RMON: Remote MONitoring

- RMON is a standard defining how remote probes or agents relay network traffic information back to a central console
- Not as prevalent as SNMP or NetFlow—supported mainly by commercial network management systems
- Cisco Network Analysis Module-2 (NAM-2), ntop (http://www.ntop.org) are examples of RMON probes
- Most RMON probes look at raw packets via SPAN/ RSPAN and generate statistics from observed traffic
- Mini-RMON statistics available on Cisco Catalyst 6500/ NAM-2, provides detailed stats from Layer 2 access ports

NAM-2 Examples



You Are Here: ◆ Monitor ▸ Overview

Overview

Current Data: as of Wed 14 May 2003, 11:07:04 Pacific

☑ Auto Refresh

Data Source: ALL SPAN

Current Data: as of Fri 16 May 2003, 12:27:27 Pacific

Retries	0
Response Time Distribution (msec)	
🟦 Responses < 5	0
🟧 Responses between 5 and 15	0
🟥 Responses between 15 and 50	0
🟩 Responses between 50 and 100	0
🟪 Responses between 100 and 200	1
🟨 Responses between 200 and 500	0
▨ Responses between 500 and 3000	0
▨ Responses > 3000	0

ay 2003, 14:22:50 Pacific

● Current Rates ○ TopN Chart ○ Cumulative Data

Port Name: Filter Clear

Showing 1-13 of 13 records

			...ation	Dropped Events/s	Bytes/s	Packets/s	Broadcast/s	Multicast/s	CRC Align Errors/s	Undersize/s	Oversize/s	Fragments/s	Jabbers/s	Collisions/s
			0.51	0.00	1118995.57	7459.23	6.52	6.00	0.00	0.00	0.00	0.00	0.00	0.00
○	2.	1/1	0.50	0.00	1091637.70	7460.63	0.00	1.07	0.00	0.00	0.00	0.00	0.00	0.00
○	3.	3/37	0.05	0.00	507.77	6.00	2.62	3.38	0.00	0.00	0.00	0.00	0.00	0.00
○	4.	3/6	0.01	0.00	3111.78	13.48	0.00	1.02	0.00	0.00	0.00	0.00	0.00	0.00
○	5.	15/1	0.01	0.00	8260.25	60.42	0.00	30.52	0.00	0.00	0.00	0.00	0.00	0.00
○	6.	3/4	0.01	0.00	1367.98	7.87	0.00	1.02	0.00	0.00	0.00	0.00	0.00	0.00
○	7.	3/17	0.00	0.00	380.05	4.27	2.32	1.80	0.00	0.00	0.00	0.00	0.00	0.00
○	8.	3/18	0.00	0.00	380.05	4.27	2.32	1.80	0.00	0.00	0.00	0.00	0.00	0.00
○	9.	2/1	0.00	0.00	3469.28	39.47	3.30	32.97	0.00	0.00	0.00	0.00	0.00	0.00
○	10.	2/2	0.00	0.00	3032.98	38.10	3.30	32.97	0.00	0.00	0.00	0.00	0.00	0.00
○	11.	2/4	0.00	0.00	3025.37	37.53	3.30	33.03	0.00	0.00	0.00	0.00	0.00	0.00
○	12.	2/3	0.00	0.00	3020.48	37.68	3.30	32.97	0.00	0.00	0.00	0.00	0.00	0.00
○	13.	2/8	0.00	0.00	2942.18	37.35	3.30	32.90	0.00	0.00	0.00	0.00	0.00	0.00

Rows per page: 15

Go to page: 1 of 1 Go

Select an item then take an action -->

Details Real-Time Report

Source: Cisco Systems, Inc.

Syslog

- De facto logging standard for hosts, network infrastructure devices, supported in all Cisco routers and switches
- Many levels of logging detail available—choose the level(s) which are appropriate for each device/situation
- ACL logging is generally contraindicated due to CPU overhead—NetFlow provides more information, doesn't max the box
- Can be used in conjunction with Anycast and databases such as MySQL (http://www.mysql.com) to provide a scalable, robust logging infrastructure
- Different facility numbers allows for segregation of log information based upon device type, function, other criteria
- Syslog-ng from http://www.balabit.com/products/syslog_ng/ adds a lot of useful functionality

Packet Capture

- Sometimes, there's just no substitute for looking at the packets on the wire
- SPAN/RSPAN/ERSPAN allow packet capture from Cisco Catalyst switches; ip packet export allows packet capture from routers
- Open source tools such as tcpdump, snoop, Wireshark (http://www.wireshark.org) on free *NIX or Windows allow inexpensive packet-capture solutions to be built and deployed
- Commercial tools such as Cisco NAM-2, NAI Sniffer/ Distributed Sniffer, Wandel and Goltermann available
- Use macroanalytical telemetry such as SNMP, NetFlow, RMON to guide your use of microanalytical telemetry (i.e., packet capture)

NetFlow for Security Purposes

NetFlow Origination

- Developed by Darren Kerr and Barry Bruins at Cisco Systems in 1996
- Primary network accounting technology in the industry
- Emerging standard traffic engineering/capacity planning technology
- Primary network anomaly-detection technology
- Answers questions regarding IP traffic:

    Who

    What

    Where

    When

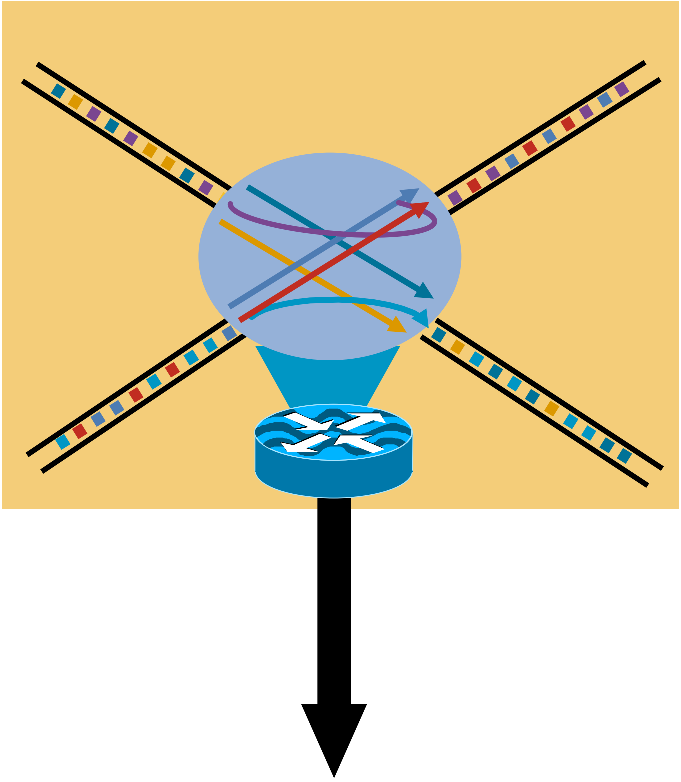
    How

    What cryptologists call “traffic analysis”

What Is a Flow?

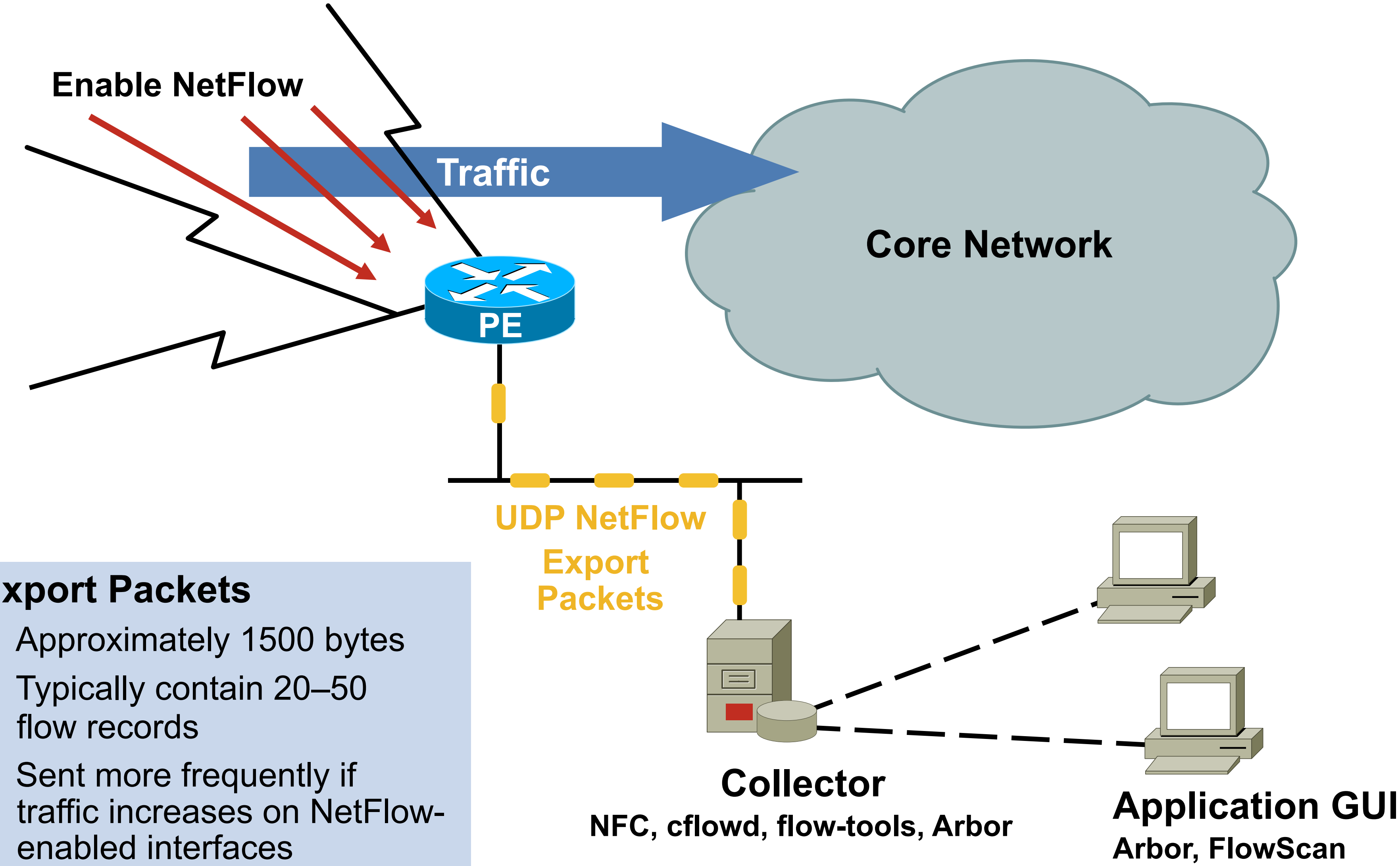
Defined by Seven Unique Keys:

- Source IP address
- Destination IP address
- Source port
- Destination port
- Layer 3 protocol type
- TOS byte (DSCP)
- Input logical interface (ifIndex)



Exported Data

Creating Export Packets



Export Packets

- Approximately 1500 bytes
- Typically contain 20–50 flow records
- Sent more frequently if traffic increases on NetFlow-enabled interfaces

Uses of NetFlow

Service Provider	Enterprise
▪ Peering Arrangements ▪ SLA VPN User Reporting ▪ Usage-Based Billing ▪ DoS/Worm Detection ▪ Traffic Engineering ▪ Troubleshooting	▪ Internet Access Monitoring (Protocol Distribution, Traffic Origin/ Destination) ▪ Associate Cost of IT to Departments ▪ More Scalable Than RMON ▪ DoS/Worm Detection ▪ Policy Compliance Monitoring ▪ Troubleshooting

Key Concept: NetFlow Scalability

- Packet capture is like a wiretap
- NetFlow is like a phone bill
- This level of granularity allows NetFlow to scale for very large amounts of traffic
- We can learn a lot from studying the phone bill
- Who’s talking to whom, over what protocols and ports, for how long, at what speed, for what duration, etc.
- NetFlow is a form of telemetry pushed from the routers/ switches—each one can be a sensor

NetFlow Versions

NetFlow Version	Comments
1	Original
5	Standard and Most Common
7	Specific to Cisco Catalyst 6500 and 7600 Series Switches Similar to Version 5, but Does Not Include AS, Interface, TCP Flag and TOS Information
8	Choice of 11 Aggregation Schemes Reduces Resource Usage
9	Flexible, Extensible File Export Format to Enable Easier Support of Additional Fields and Technologies; Coming Out Now Are MPLS, Multicast, and BGP Next-Hop

IOS NetFlow Configuration—Version 5 Export

```
ip cef
…
interface FastEthernet0
 ip address 192.168.131.100 255.255.255.0
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 ip flow ingress
…
ip flow-export destination 192.168.131.200 2055
…
ip flow-export version 5
```

Why a New Version?

- Fixed formats (versions 1, 5, 7 and 8) are not flexible and adaptable

  Cisco needed to build a new version each time a customer wanted to export new fields

- When new versions are created, partners need to reengineer to support the new export format

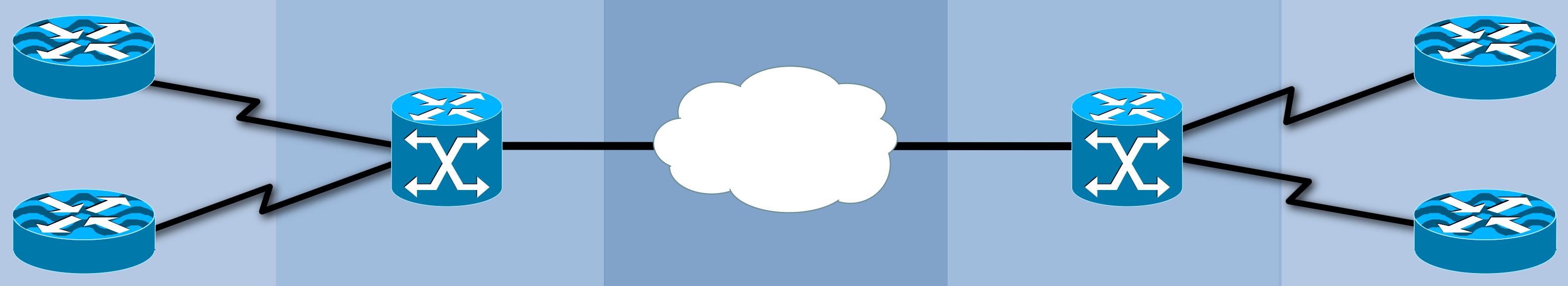
Solution: Build a Flexible and Extensible Export Format

NetFlow v9 Principles

- Version 9 is an export format
- Still a push model
- Send the template regularly (configurable)
- Independent of the underlying protocol, it is ready for any reliable protocol (i.e., TCP, SCTP)

NetFlow in the Topology

Network Layer



	Access	Distribution	Core	Distribution	Access
Applications	- Attack detection - User (IP) monitoring - Application monitoring	- Billing - Chargeback - AS peer monitoring - Attack detection	- Traffic engineering - Traffic analysis - Attack detection	- Billing - Chargeback - AS peer monitoring - Attack detection	- Attack detection - User (IP) monitoring - Application monitoring
NetFlow Features	- Aggregation schemes (v8) - "show ip cache flow" command - Arbor Networks	- NetFlow MPLS egress accounting - BGP next-hop (v9) - Arbor Networks	- MPLS Aware NetFlow (v9) - BGP Next-hop (v9) - Sampled NetFlow - Arbor Networks	- NetFlow MPLS Egress Accounting - BGP Next-hop (v9) - Arbor Networks	- Aggregation Schemes (v8) - "show ip cache flow" command - Arbor Networks

What Is an Anomaly?

- An event or condition in the network that is identified as a statistical abnormality when compared to typical traffic patterns gleaned from previously collected profiles and baselines

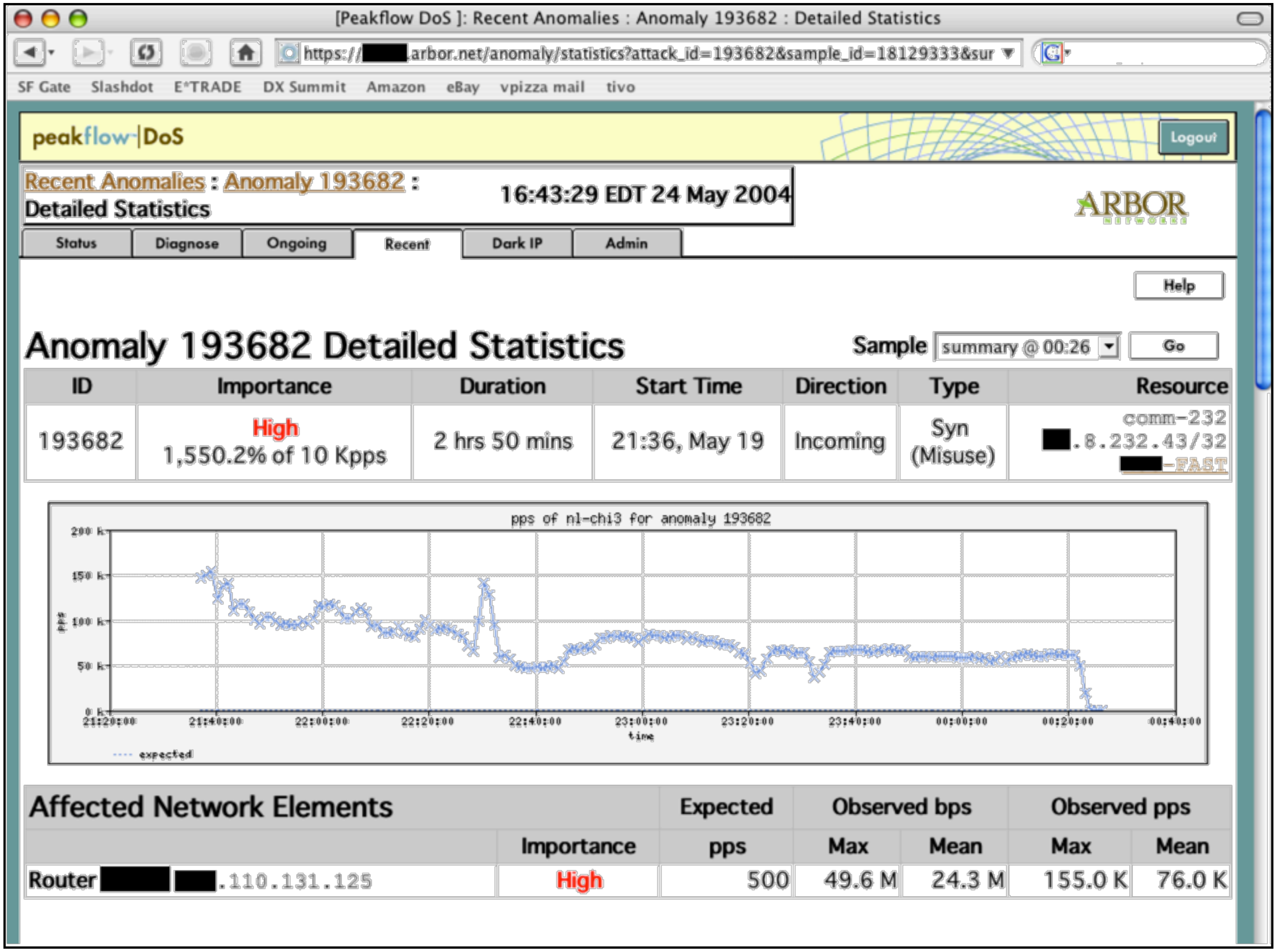
NetFlow-Based Traffic Characterization and Anomaly Detection with Arbor Networks

Network Anomaly Detection and Traffic Characterization/ Capacity Planning

peakflow™ | SP

- Most widely deployed anomaly detection system for SPs
- Uses NetFlow to quickly identify, classify, and scope DoS, worms, etc.
- Traffic component combines NetFlow traffic characterization with BGP
- Allows comprehensive peering analysis in real-time
- A “force multiplier” which greatly reduces reaction-times by providing the relevant information up-front
- Can also generate its own flows from packet-capture if NetFlow isn’t available

Anomaly Example: Detail



[Peakflow DoS]: Recent Anomalies : Anomaly 193682 : Detailed Statistics

https://[redacted].arbor.net/anomaly/statistics?attack_id=193682&sample_id=18129333&sur

Recent Anomalies : Anomaly 193682 :
Detailed Statistics

16:43:29 EDT 24 May 2004

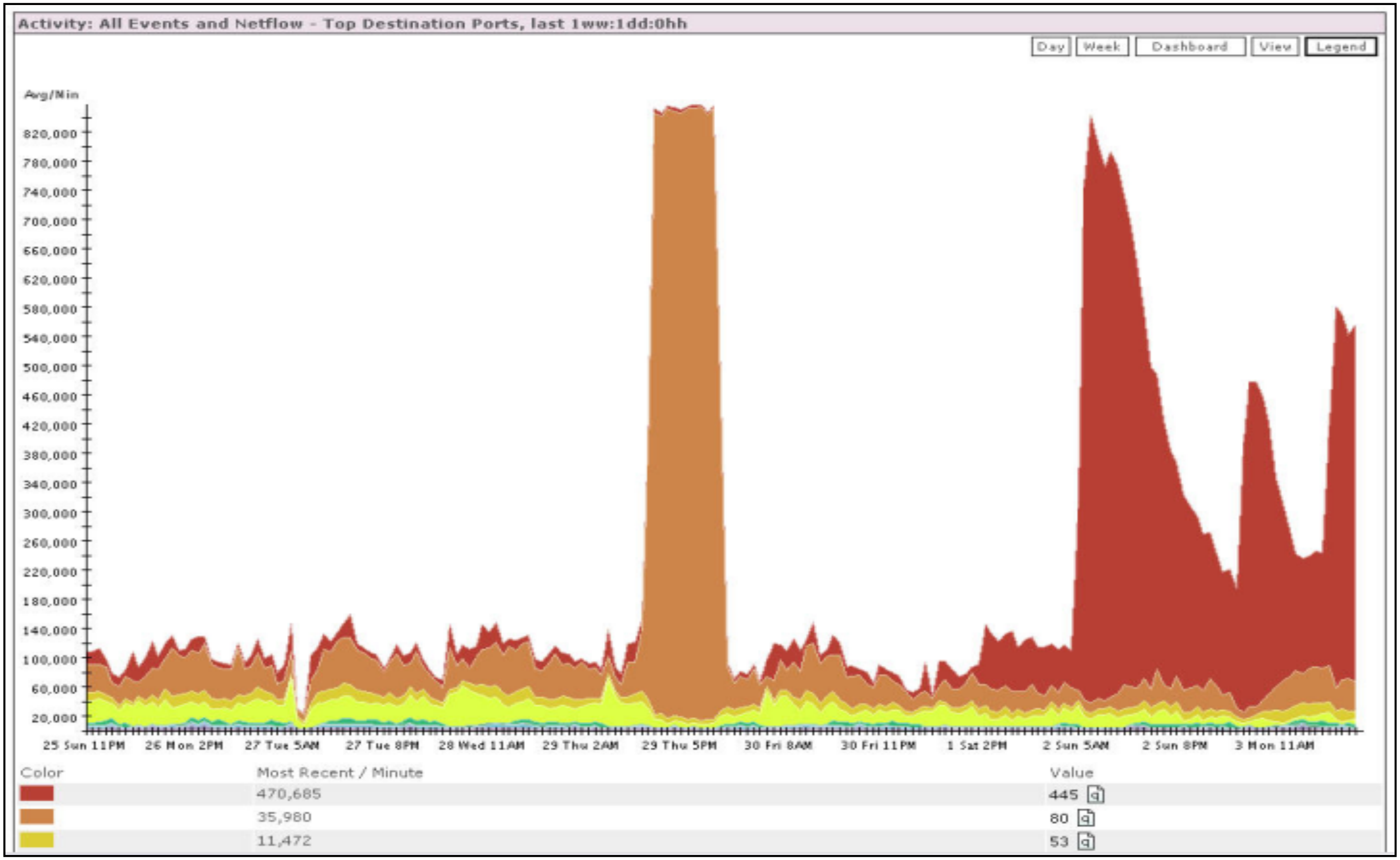
Anomaly 193682 Detailed Statistics

Sample summary @ 00:26

ID	Importance	Duration	Start Time	Direction	Type	Resource
193682	High 1,550.2% of 10 Kpps	2 hrs 50 mins	21:36, May 19	Incoming	Syn (Misuse)	comm-232 [redacted].8.232.43/32 [redacted]-FAST

Affected Network Elements		Expected	Observed bps		Observed pps	
	Importance	pps	Max	Mean	Max	Mean
Router [redacted] [redacted].110.131.125	High	500	49.6 M	24.3 M	155.0 K	76.0 K

Sasser Detection


Activity: All Events and Netflow - Top Destination Ports, last 1ww:1dd:0hh

Color	Most Recent / Minute	Value
🟥	470,685	445
🟧	35,980	80
🟨	11,472	53

Traceback Techniques

Traceback Essentials

- If source prefix is not spoofed:
  - Routing table
  - Internet Routing Registry (IRR)—whois
  - Direct site contact—ARIN, RIPE, APNIC
- If source prefix is spoofed:
  - Trace packet flow through the network
  - Find upstream connection
  - Upstream needs to continue tracing

Traceback Spoofed IPv4 Addresses

- Source: inside or outside?
- Once you have a fundamental understanding of the type of attack (source address and protocol type), you then need to trace to the ingress point
- Two main techniques:

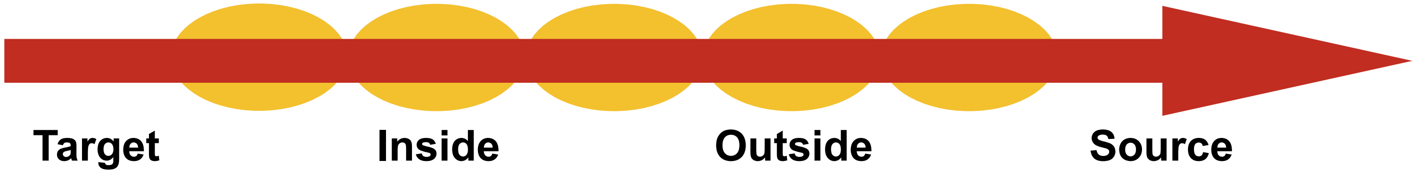
  Hop-by-hop

  Jump to ingress

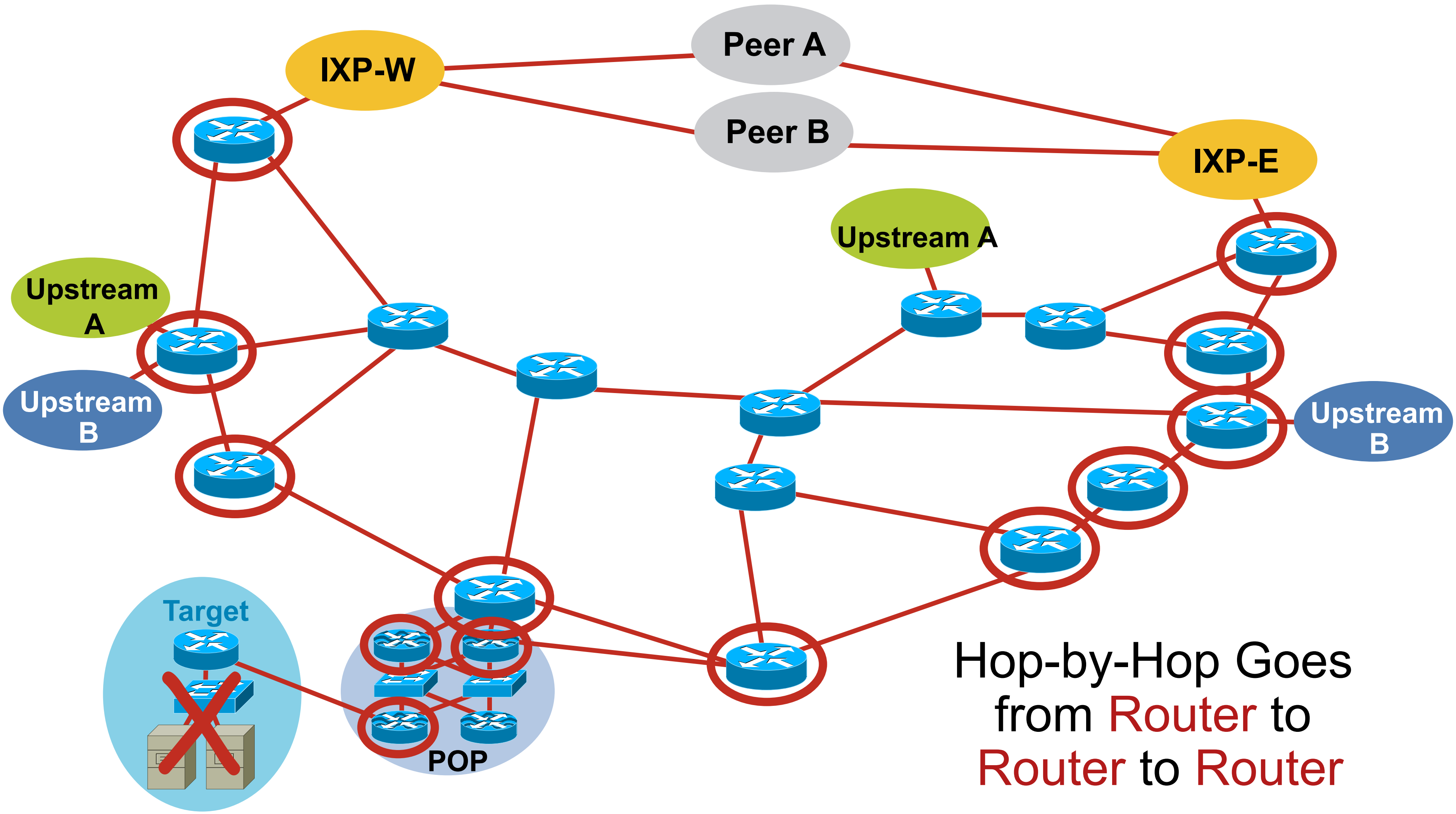
Traceback via Hop-by-Hop Technique

Hop-by-Hop Traceback Takes Time

- Starts from the beginning and traces to the source of the problem
- Needs to be done on each router
- Often requires splitting—tracing two separate paths
- Speed is the limitation of the technique



Traceback via Hop-by-Hop Technique

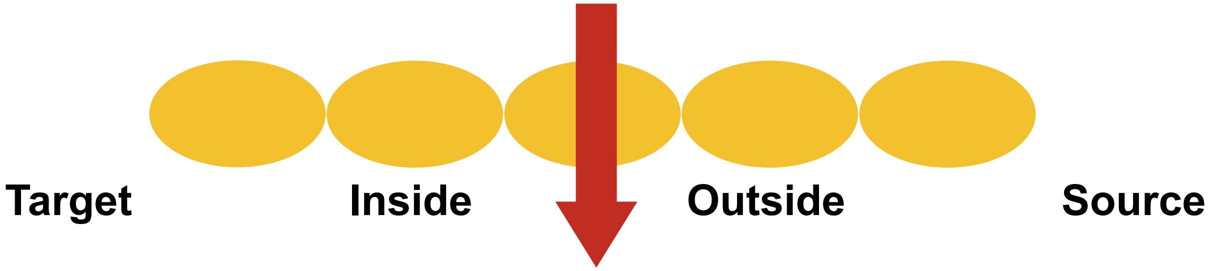


Hop-by-Hop Goes from Router to Router to Router

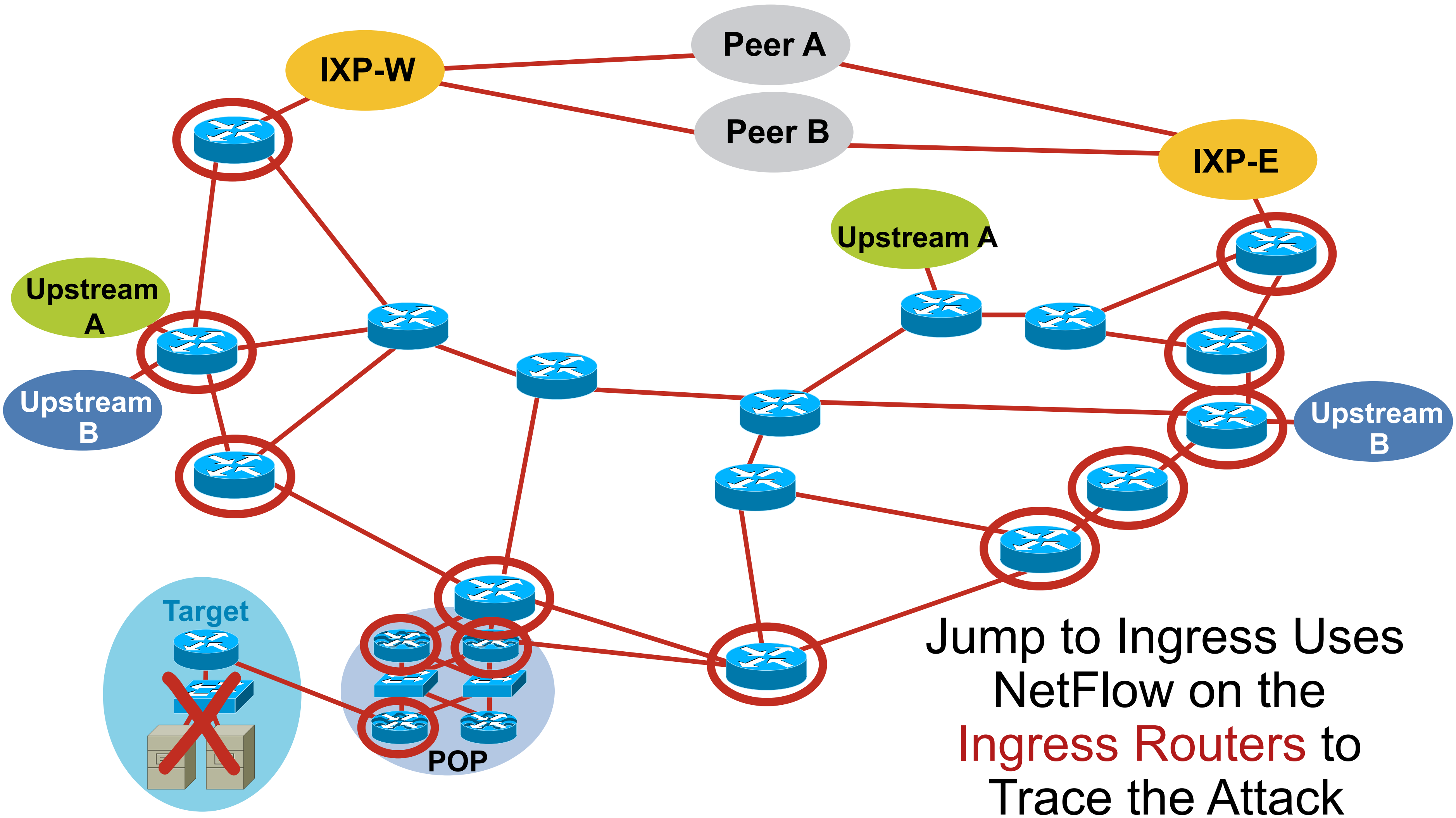
Traceback via the Jump to Ingress Technique

Jump to Ingress Tracebacks Divides the Problem in Half

- Is the attack originating from inside the network or outside the network?
- Jump to the ingress border routers to see if the attack is entering the network from the outside
- Advantage: speed—are we the source or is someone else the source?



Traceback via the Jump to Ingress Technique



Jump to Ingress Uses NetFlow on the Ingress Routers to Trace the Attack

Traceback Spoofed IPv4 Addresses

Traceback Techniques

- Apply temporary ACLs with log-input and examine the logs (like classification)
- Query NetFlow's flow table
  - Show ip cache-flow if NetFlow is enabled
- Backscatter traceback technique
- Traceback using NetFlow telemetry

Traceback with ACLs

- Original traceback technique
- Risk: inserting change into a network that is under attack
- Risk: log-input requires the forwarding ASIC to punt the packet to capture log information
- BCP is to apply the filter, capture just enough information, then remove the filter

Traceback with ACLs

```
 access-list 170 permit icmp any any echo
 access-list 170 permit icmp any any echo-reply log-input
 access-list 170 permit udp any any eq echo
 access-list 170 permit udp any eq echo any
 access-list 170 permit tcp any any established
 access-list 170 permit tcp any any
 access-list 170 permit ip any any

 interface serial 0
   ip access-group 170 out
! Wait a short time - (i.e 10 seconds)
   no ip access-group 170 out
```

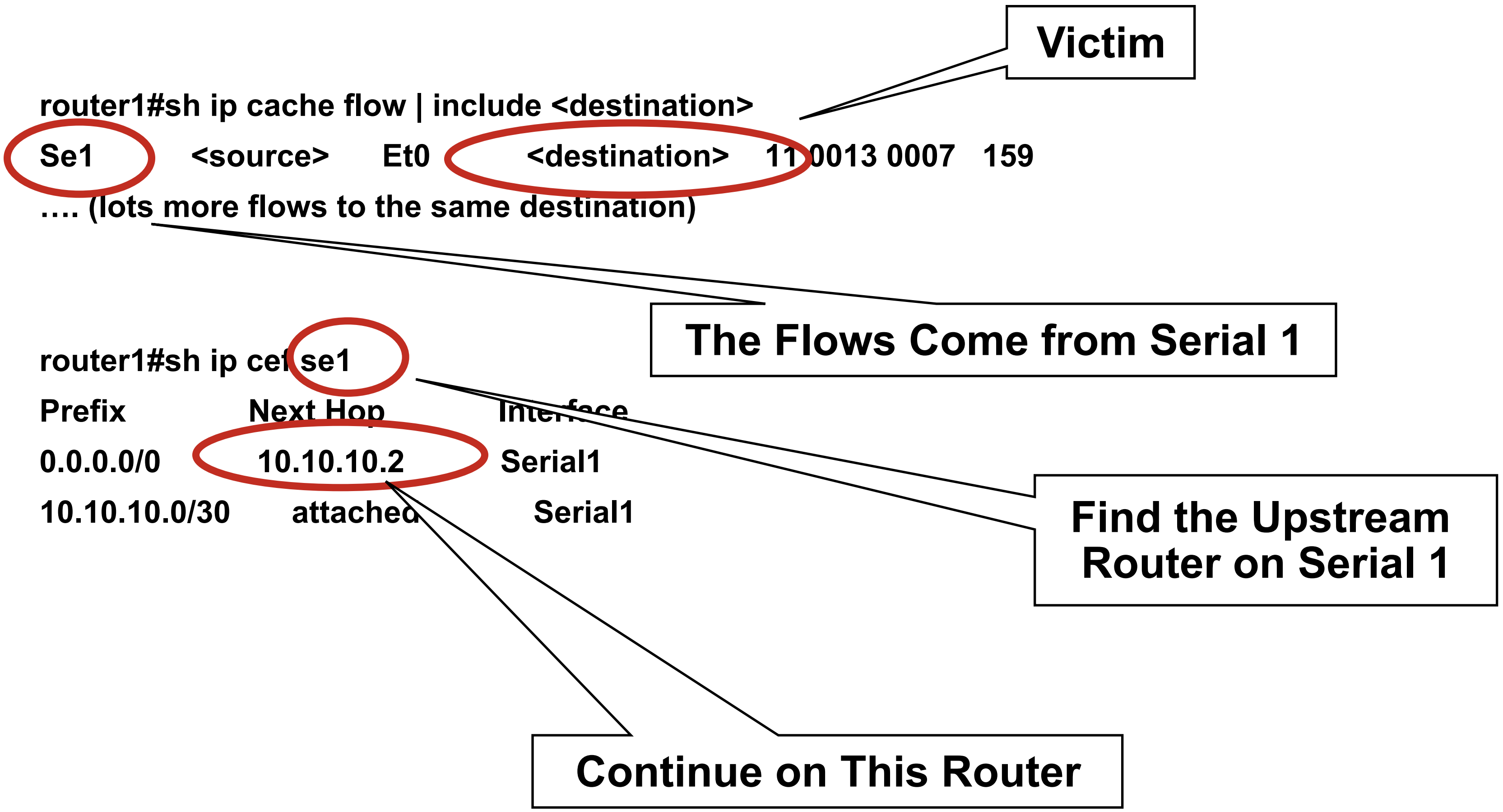
Traceback with ACLs Output

- Validate the capture with show access-list 170; make sure it the packets we counted
- View the log with show logging for input interface:

```
%SEC-6-IPACCESSLOGDP: list 170 permit icmp 192.168.212.72
(Serial0 *HDLC*) -> 172.19.61.10 (0/0), 1 packet

%SEC-6-IPACCESSLOGDP: list 170 permit icmp 172.16.132.154
(Serial0 *HDLC*) -> 172.19.61.10 (0/0), 1 packet

%SEC-6-IPACCESSLOGDP: list 170 permit icmp 192.168.45.15
(Serial0 *HDLC*) -> 172.19.61.10 (0/0), 1 packet

%SEC-6-IPACCESSLOGDP: list 170 permit icmp 192.168.45.142
(Serial0 *HDLC*) -> 172.19.61.10  (0/0), 1 packet

%SEC-6-IPACCESSLOGDP: list 170 permit icmp 172.16.132.47
(Serial0 *HDLC*) -> 172.19.61.10 (0/0), 1 packet
```

Netflow Traceback Techniques

Traceback with NetFlow



```
router1#sh ip cache flow | include <destination>
Se1      <source>     Et0         <destination>     11 0013 0007   159
…. (lots more flows to the same destination)

router1#sh ip cef se1
Prefix            Next Hop          Interface
0.0.0.0/0           10.10.10.2      Serial1
10.10.10.0/30       attached            Serial1
```

Victim

The Flows Come from Serial 1

Find the Upstream Router on Serial 1

Continue on This Router

show ip cache flow

```
router_A#sh ip cache flow
IP packet size distribution (85435 total packets):
   1-32   64   96  128  160  192  224  256  288  320  352  384  416  448  480
   .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000

    512  544  576 1024 1536 2048 2560 3072 3584 4096 4608
   .000 .000 .000 .000 1.00 .000 .000 .000 .000 .000 .000

IP Flow Switching Cache, 278544 bytes
  2728 active, 1368 inactive, 85310 added
  463824 ager polls, 0 flow alloc failures
  Active flows timeout in 30 minutes
  Inactive flows timeout in 15 seconds
  last clearing of statistics never
Protocol         Total    Flows   Packets Bytes  Packets Active(Sec) Idle(Sec)
--------         Flows     /Sec     /Flow  /Pkt     /Sec     /Flow     /Flow
TCP-X                2      0.0         1  1440      0.0       0.0       9.5
TCP-other        82580     11.2         1  1440     11.2       0.0      12.0
Total:           82582                              11.2       0.0      12.0

SrcIf         SrcIPaddress    DstIf         DstIPaddress    Pr SrcP DstP  Pkts
Et0/0         132.122.25.60   Se0/0         192.168.1.1     06 9AEE 0007     1
Et0/0         139.57.220.28   Se0/0         192.168.1.1     06 708D 0007     1
Et0/0         165.172.153.65  Se0/0         192.168.1.1     06 CB46 0007     1
```

Protocol

Flow Information Summary

Flow Details

Useful NetFlow CLI Tricks

- Router>show ip cache flow | include <ip address>

  Determine flows pertaining to a specific victim or attacker

- Router>show ip cache flow | include _1$

  Determine single packet flows (potential scanning flows)

- Router>show ip cache flow | include K|M$

  Determine really large flows (in 1,000s or 1,000,000s of packets)

- Router>show ip cache flow | include <protocol / port>

  Determine flows with specific protocols/ports

Traceback with NetFlow Example
Tracing W32.Blaster Infected Hosts

W32.Blaster-Infected Hosts Attempt to Replicate to Random Systems Using Port 135, Which Is Hex 0087

```
Router>show ip cache flow | include 0087
:
SrcIf SrcIPaddress DstIf DstIPaddress Pr SrcP DstP Pkts
Fa2/0 XX.XX.XX.242 Fa1/0 XX.XX.XX.119 06 0B88 0087 1
Fa2/0 XX.XX.XX.242 Fa1/0 XX.XX.XX.169 06 0BF8 0087 1
Fa2/0 XX.XX.XX.204 Fa1/0 XX.XX.XX.63  06 0E80 0087 1
Fa2/0 XX.XX.XX.204 Fa1/0 XX.XX.XX.111 06 0CB0 0087 1
Fa2/0 XX.XX.XX.204 Fa1/0 XX.XX.XX.95  06 0CA0 0087 1
Fa2/0 XX.XX.XX.204 Fa1/0 XX.XX.XX.79  06 0C90 0087 1
```

Traceback with NetFlow Telemetry

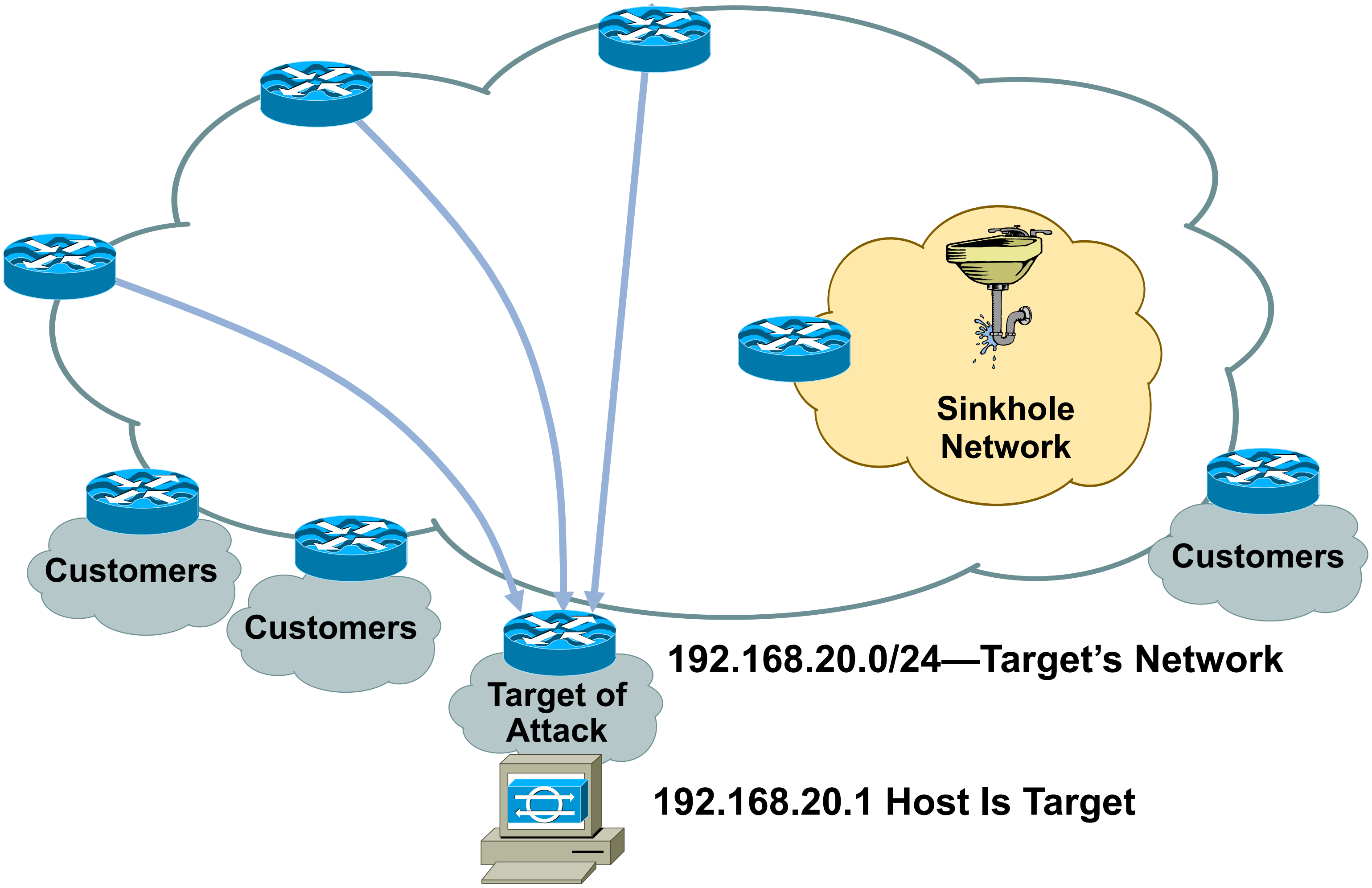
- Routers on the edge of the network can export NetFlow data reporting detailed traffic flow information
- This telemetry can be processed to detect anomalies and to traceback the attack to the source(s)
- Open source and commercial products available
- Arbor PeakFlow provides one example that has operationally proven its value

Attract and Analyze: Sinkholes

Sinkhole Routers/Networks

- Sinkholes are a topological security feature—think network honeypot
- Router or workstation built to suck in traffic and assist in analyzing attacks (original use)
- Redirect attacks away from the customer—working the attack on a router built to withstand the attack
- Used to monitor attack noise, scans, data from misconfiguration and other activity (via the advertisement of default or unused IP space)
- Traffic is typically diverted via BGP route advertisements and policies
- Leverage instrumentation in a controlled environment
  - Pull the traffic past analyzers/analysis tools

Sinkhole Routers/Networks



Sinkhole Network

Customers

Customers

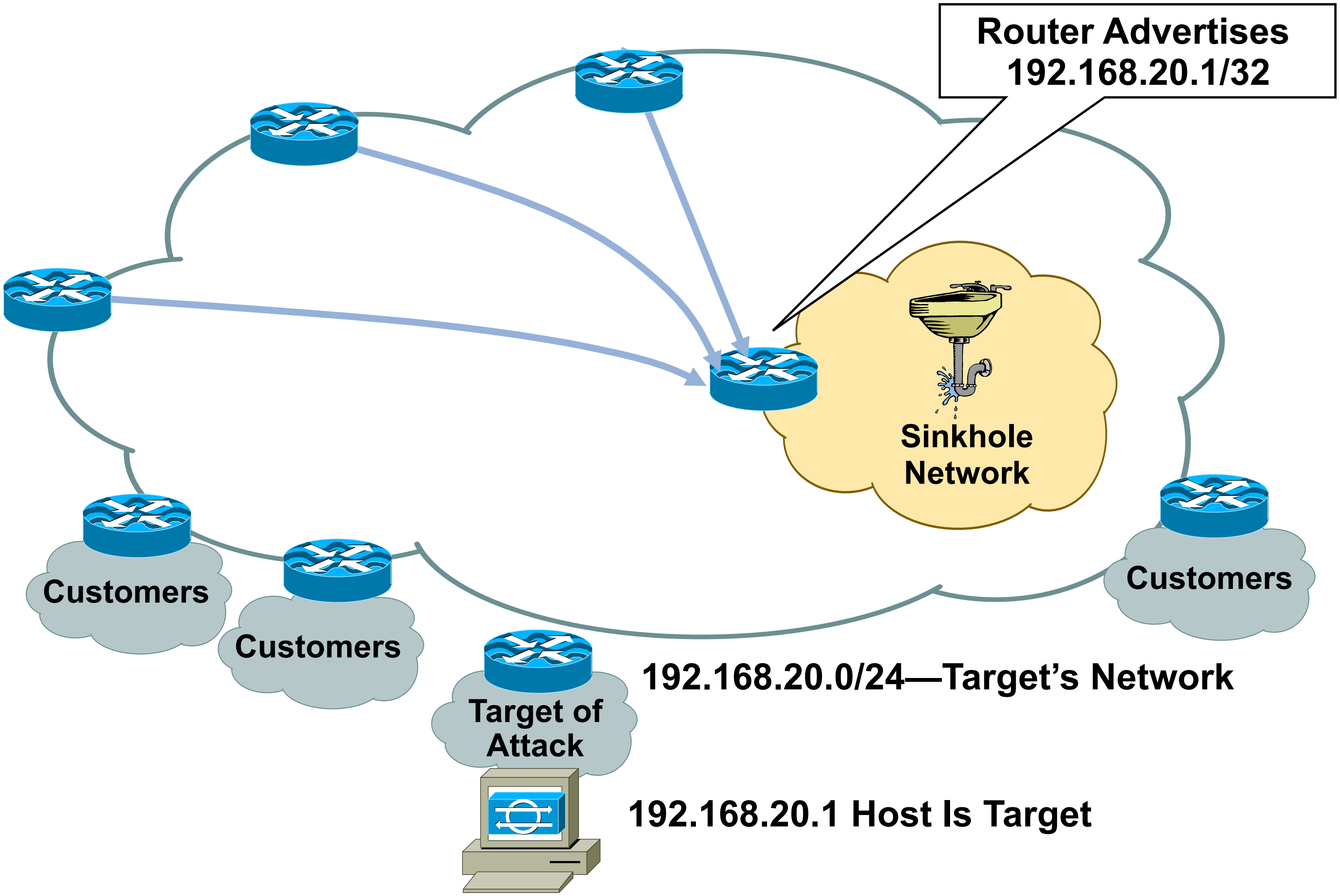
Customers

Target of Attack

192.168.20.0/24—Target's Network

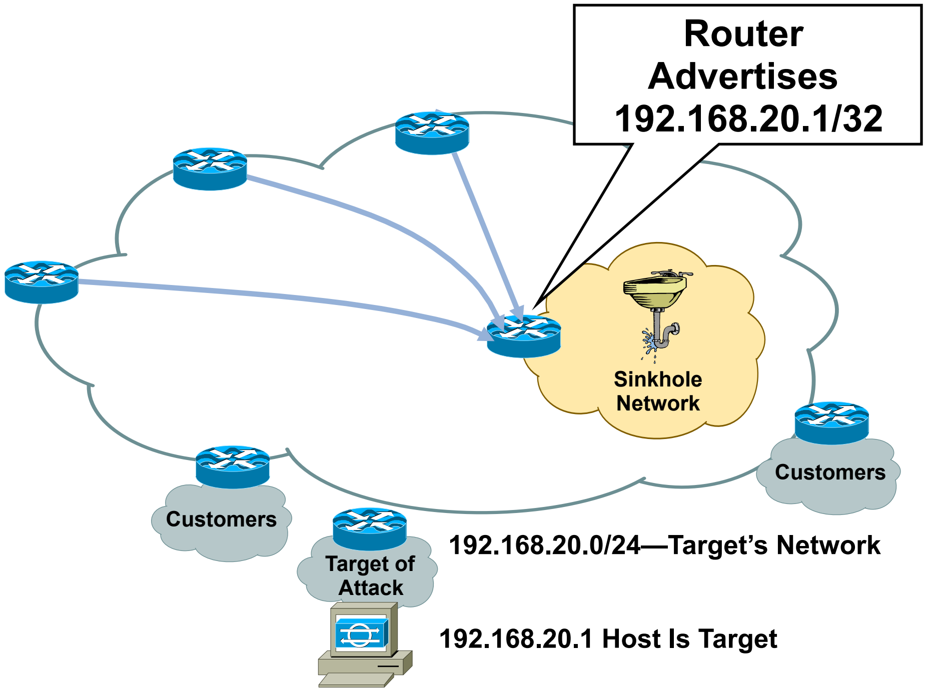
192.168.20.1 Host Is Target

Sinkhole Routers/Networks


Router Advertises 192.168.20.1/32
Sinkhole Network
Customers
Customers
Customers
Target of Attack
192.168.20.0/24—Target's Network
192.168.20.1 Host Is Target

Sinkhole Routers/Networks

- Attack is pulled away from customer/aggregation router
- Can now apply classification ACLs, packet capture, etc.
- Objective is to minimize the risk to the network while investigating the attack incident


Router Advertises 192.168.20.1/32
Sinkhole Network
Customers
Customers
Target of Attack
192.168.20.0/24—Target's Network
192.168.20.1 Host Is Target

Sinkhole Routers/Networks

- Advertising "space" from the sinkhole will pull down all sorts of garbage (and potentially interesting) traffic:

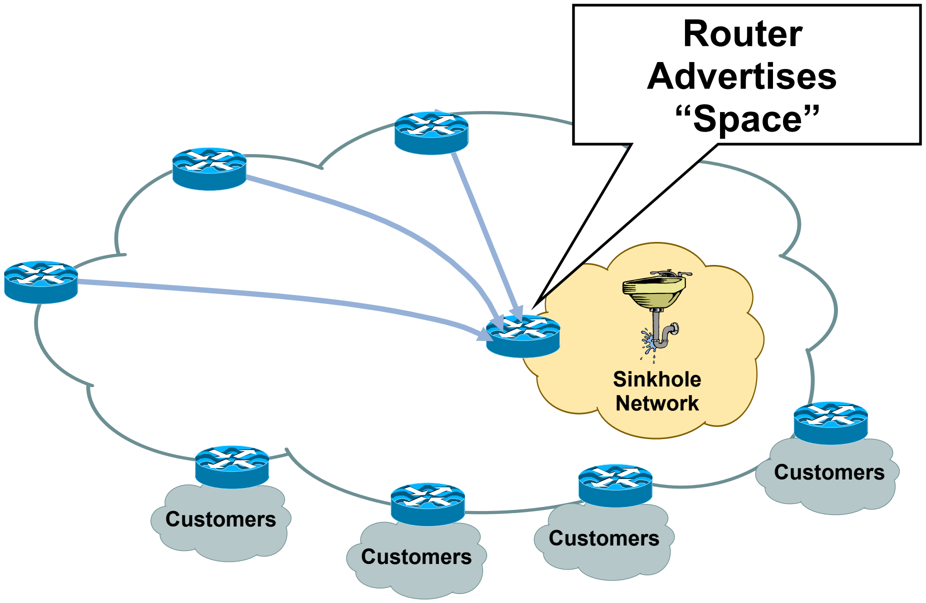
  Customer traffic when circuits flap

  Network scans to unallocated address space

  Worm traffic

  Backscatter

- Place tracking tools in the sinkhole network to monitor the noise



What to Monitor in a Sinkhole?

- Scans on dark IP (allocated and announced but unassigned address space)

  Who is scoping out the network—pre-attack planning, worms

- Scans on bogons (unallocated)

  Worms, infected machines, and Bot creation

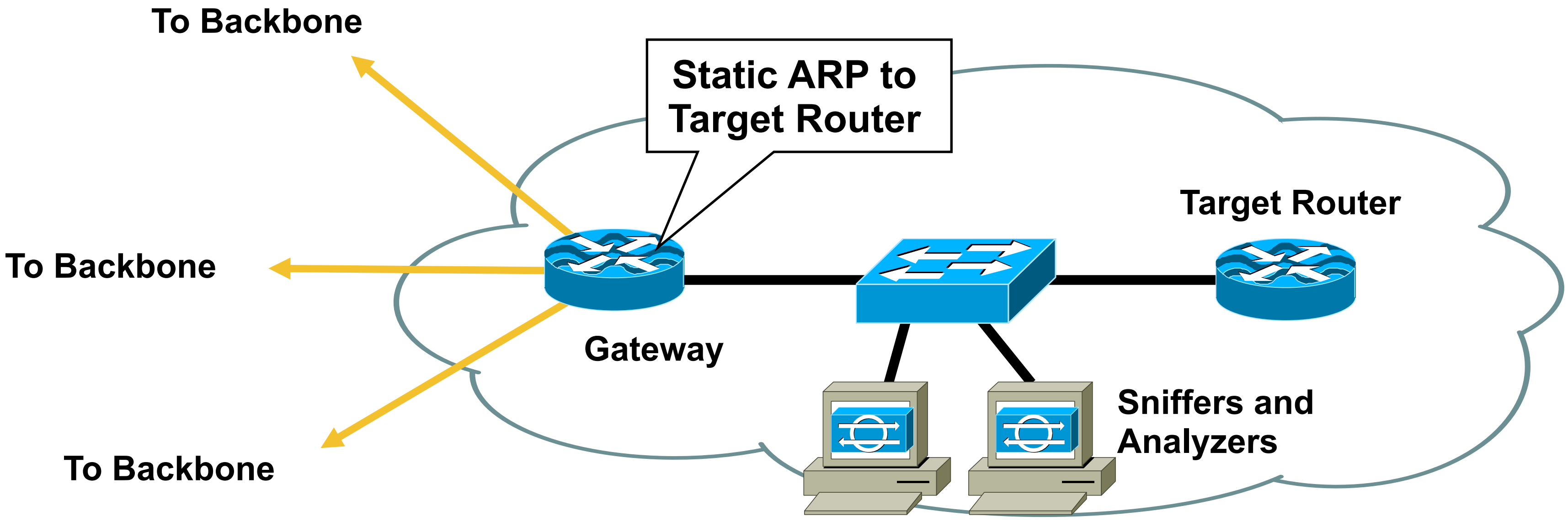
- Backscatter from attacks

  Who is getting attacked

- Backscatter from garbage traffic (RFC-1918 leaks)

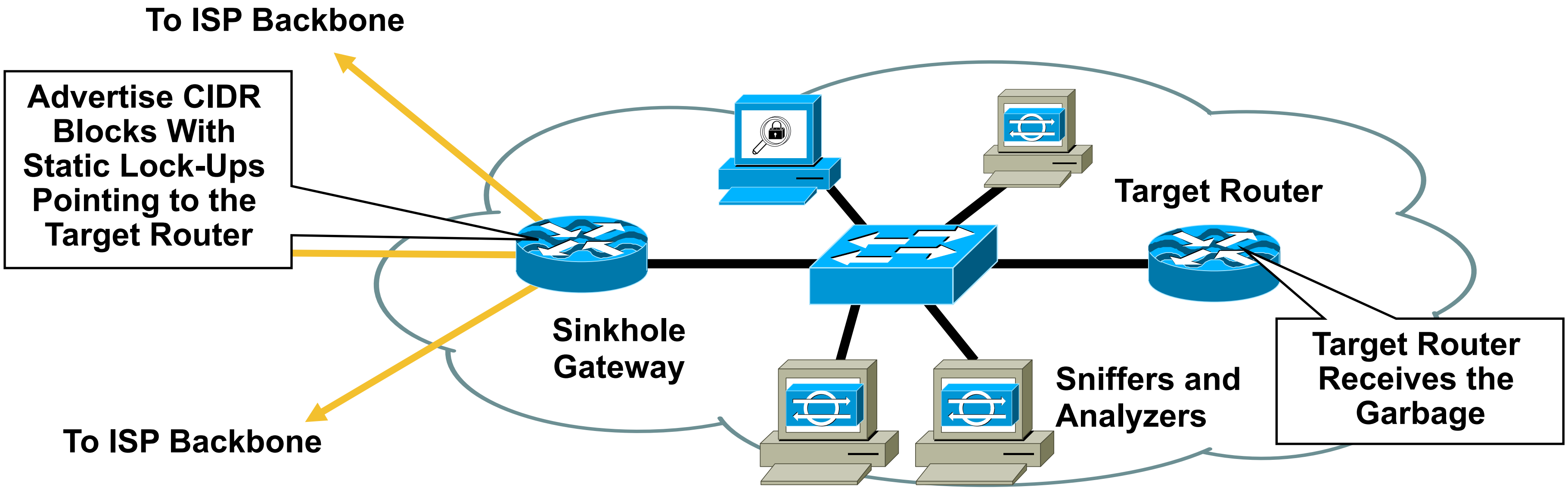
  Which customers have misconfiguration or “leaking” networks

Sinkhole Architecture



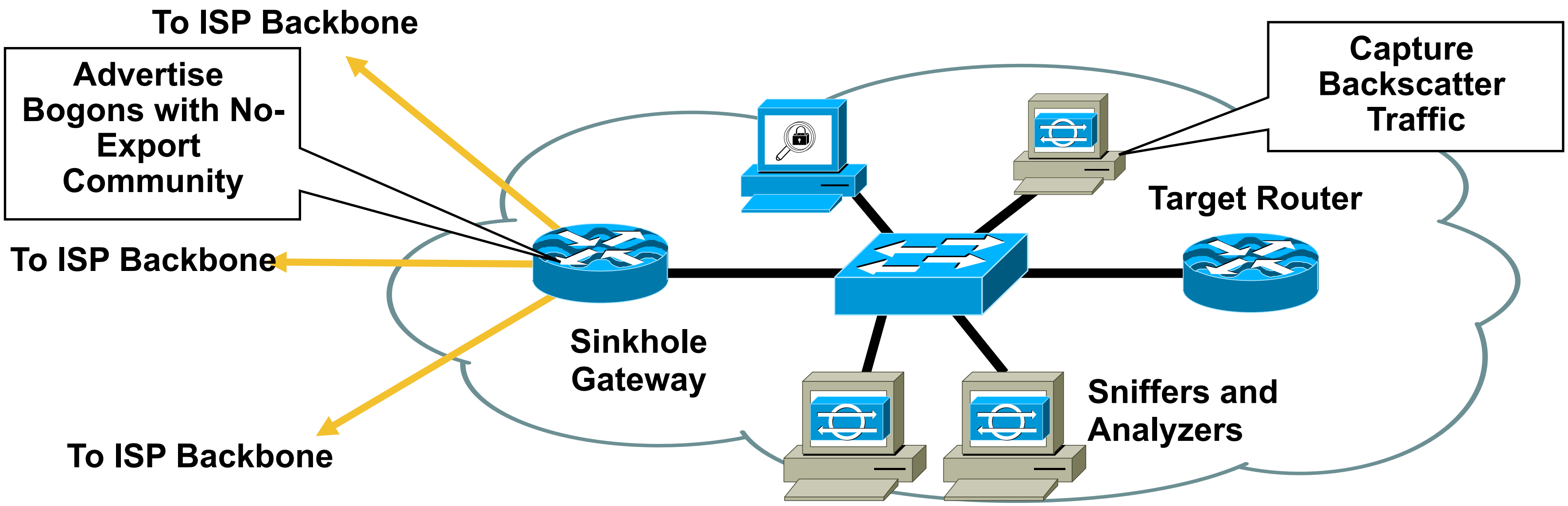
- Expand sinkhole with dedicated router into a variety of tools
- Pull DDoS attack to the sinkhole and forward data toward target router
- Static ARP to the target router keeps the sinkhole operational—target router can crash from attack and static ARP will keep gateway forwarding traffic to the Ethernet switch—rather than generating lots of ICMP error messages
- Observe trends and deviations, reserve packet detail for research and specific analysis

Sinkholes: Advertising Dark IP


To ISP Backbone
Advertise CIDR Blocks With Static Lock-Ups Pointing to the Target Router
Target Router
Sinkhole Gateway
Sniffers and Analyzers
Target Router Receives the Garbage
To ISP Backbone

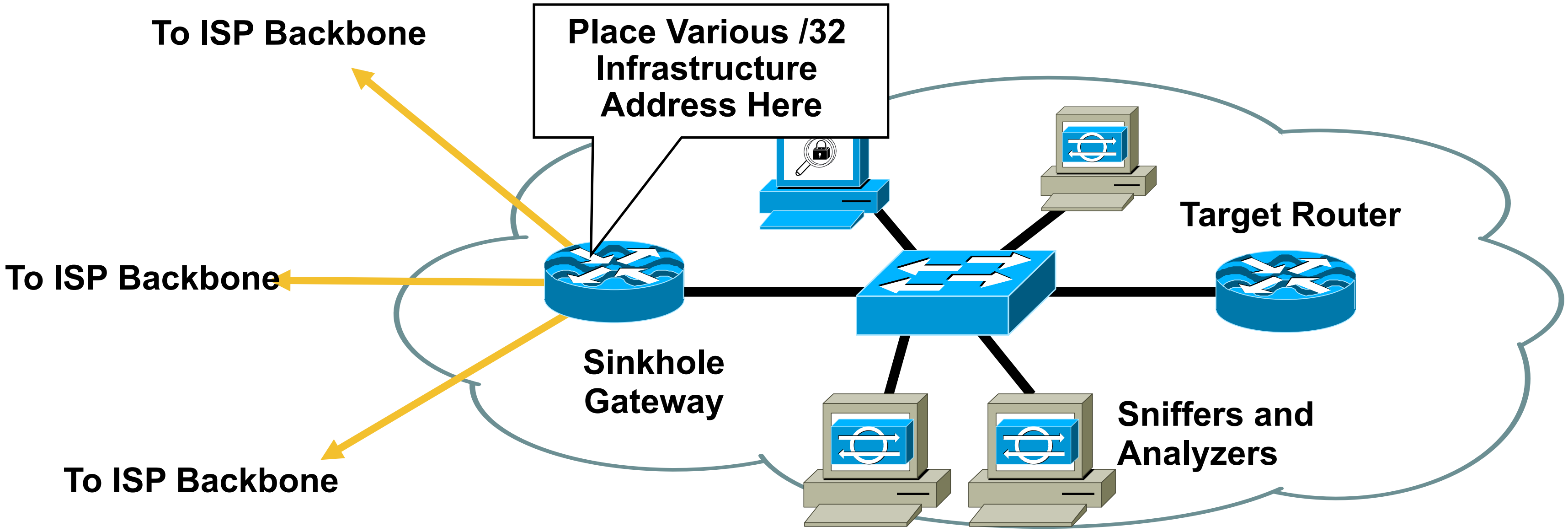
- Move the CIDR Block Advertisements (or at least more-specifics of those advertisements) to sinkholes
- Does not impact BGP routing—route origination can happen anywhere in the iBGP mesh (careful about MEDs and aggregates)
- Control where you drop the packet
- Turns networks inherent behaviors into a security tool

Monitoring Backscatter



- Advertise bogon blocks with NO_EXPORT community and an explicit safety community (plus prefix-based egress filtering on the edge)
- Static/set the BGP NEXT_HOP for the bogon to a backscatter collector workstation (as simple as TCPdump)
- Pulls in backscatter for that range—allows monitoring

Monitoring Scan Rates

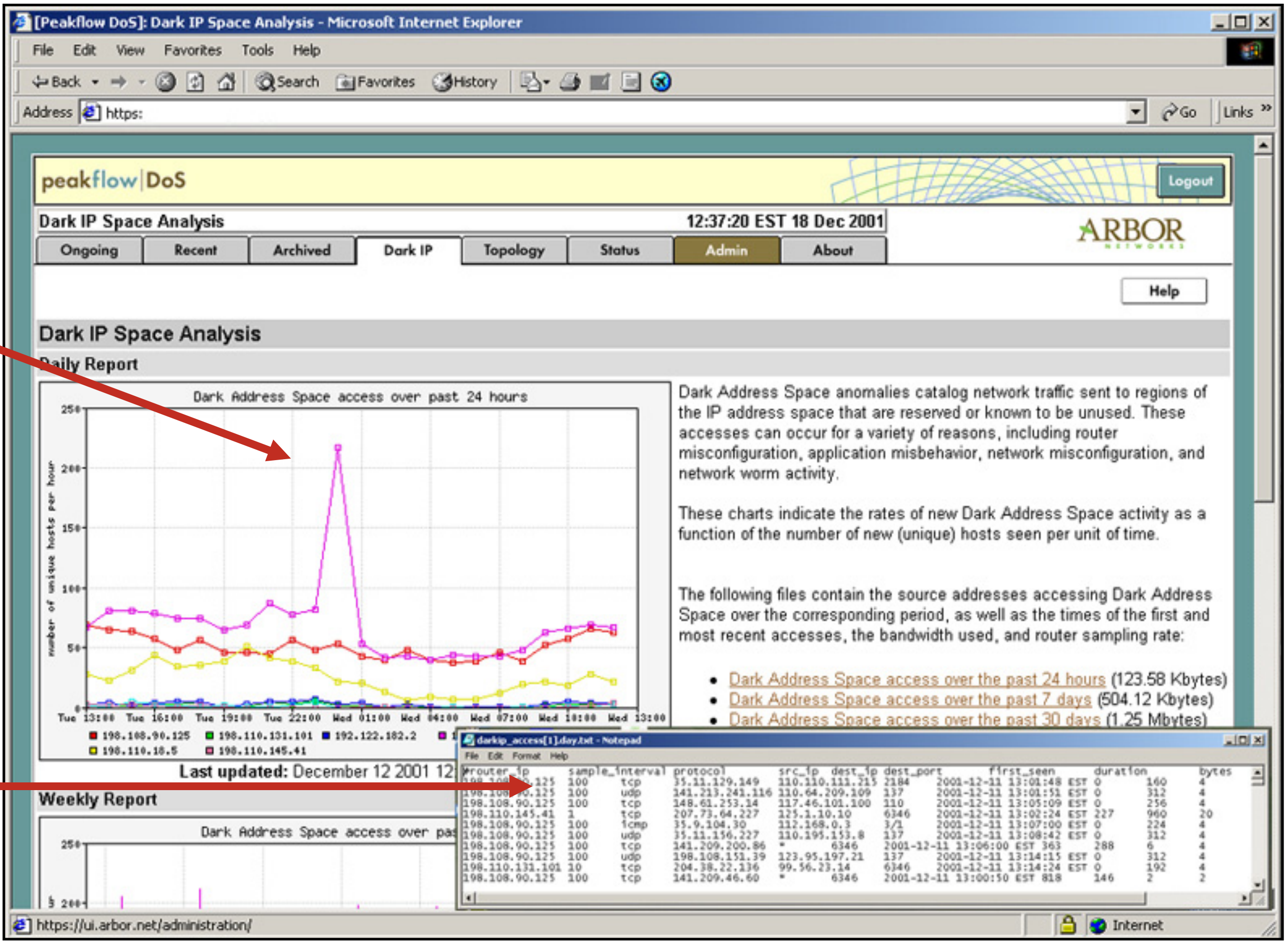


- Select /32 (or larger) address from different block of your address space; advertise them out the sinkhole
- Assign them to a workstation built to monitor and log scans (Arbor Network's Dark IP PeakFlow module is one turnkey commercial tool that can monitor scan rates via data collected from the network)

Worm Detection and Reporting UI

Operator Instantly Notified of Worm Infection

System Automatically Generates a List of Infected Hosts for Quarantine and Cleanup



Dark IP Space Analysis 12:37:20 EST 18 Dec 2001

Dark IP Space Analysis

Daily Report

Dark Address Space anomalies catalog network traffic sent to regions of the IP address space that are reserved or known to be unused. These accesses can occur for a variety of reasons, including router misconfiguration, application misbehavior, network misconfiguration, and network worm activity.

These charts indicate the rates of new Dark Address Space activity as a function of the number of new (unique) hosts seen per unit of time.

The following files contain the source addresses accessing Dark Address Space over the corresponding period, as well as the times of the first and most recent accesses, the bandwidth used, and router sampling rate:

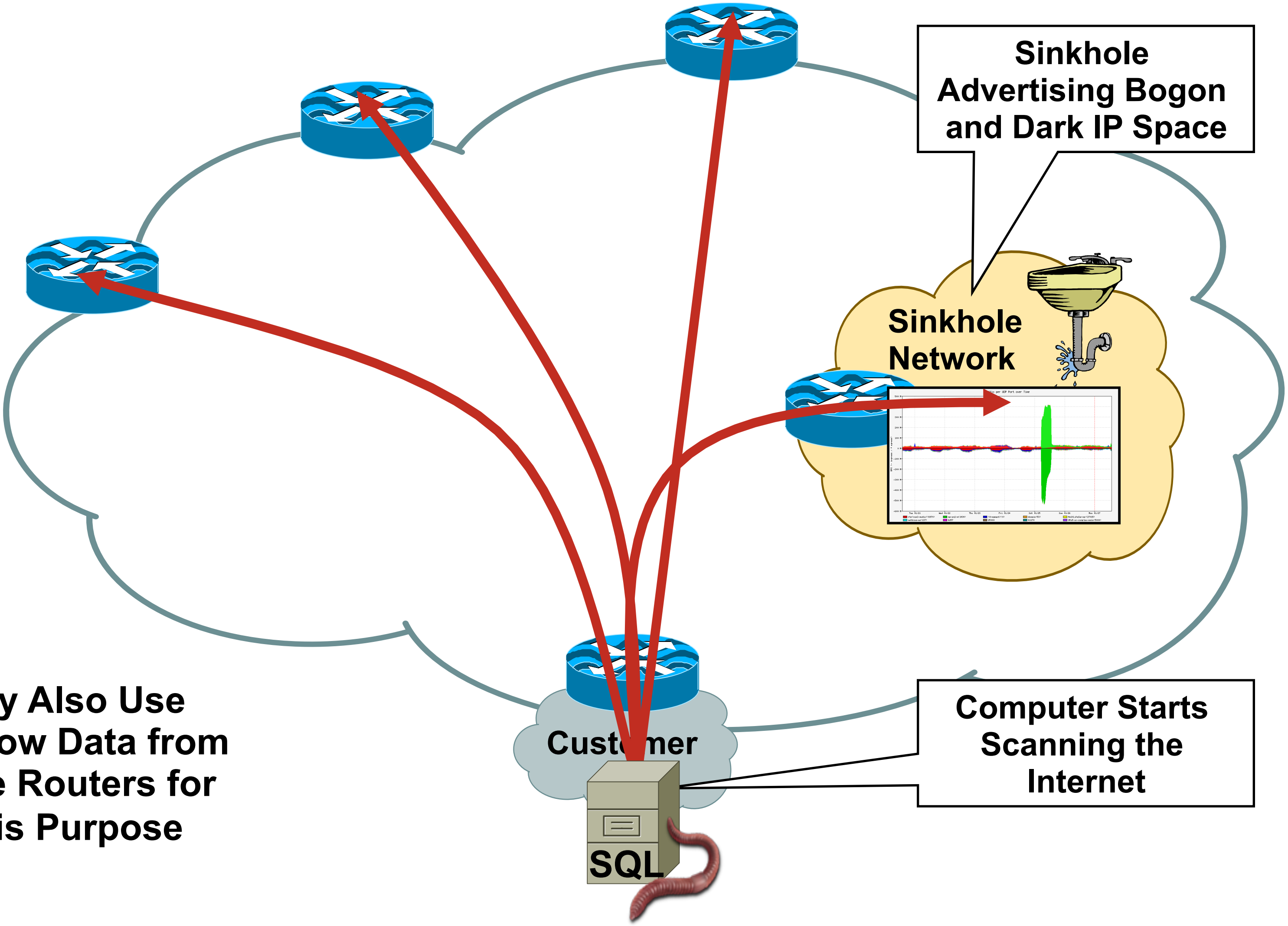
- Dark Address Space access over the past 24 hours (123.58 Kbytes)
- Dark Address Space access over the past 7 days (504.12 Kbytes)
- Dark Address Space access over the past 30 days (1.25 Mbytes)

Last updated: December 12 2001 12

Weekly Report

```
#router_ip      sample_interval protocol  src_ip          dest_ip        dest_port      first_seen              duration       bytes
198.108.90.125  100     tcp     35.11.129.149   110.110.111.215 2184    2001-12-11 13:01:48 EST 0       160     4
198.108.90.125  100     udp     141.213.241.116 110.64.209.109  137     2001-12-11 13:01:51 EST 0       312     4
198.108.90.125  100     tcp     148.61.253.14   117.46.101.100  110     2001-12-11 13:05:09 EST 0       256     4
198.110.145.41  1       tcp     207.73.64.227   125.1.10.10     6346    2001-12-11 13:02:24 EST 227     960     20
198.108.90.125  100     icmp    35.9.104.30     112.168.0.3     3/1     2001-12-11 13:07:00 EST 0       224     4
198.108.90.125  100     udp     35.11.156.227   110.195.153.8   137     2001-12-11 13:08:42 EST 0       312     4
198.108.90.125  100     tcp     141.209.200.86  *       6346    2001-12-11 13:06:00 EST 363     288     6       4
198.108.90.125  100     udp     198.108.151.39  123.95.197.21   137     2001-12-11 13:14:15 EST 0       312     4
198.110.131.101 10      tcp     204.38.22.136   99.56.23.14     6346    2001-12-11 13:14:24 EST 0       192     4
198.108.90.125  100     tcp     141.209.46.60   *       6346    2001-12-11 13:00:50 EST 818     146     2       2
```

Sinkholes: Worm Detection



Sinkhole Advertising Bogon and Dark IP Space

Computer Starts Scanning the Internet

May Also Use NetFlow Data from Edge Routers for This Purpose

But I’m Not a Core Provider?

- All networks aggregate traffic somewhere

  Control where and how, control your traffic, not vice versa

- Default route is a “strange attractor”

  Do you use a default route? Congratulations, you have a sinkhole

  Don’t let those packets drop in vain

- Collect data about the traffic and realize the benefits of sinkholes

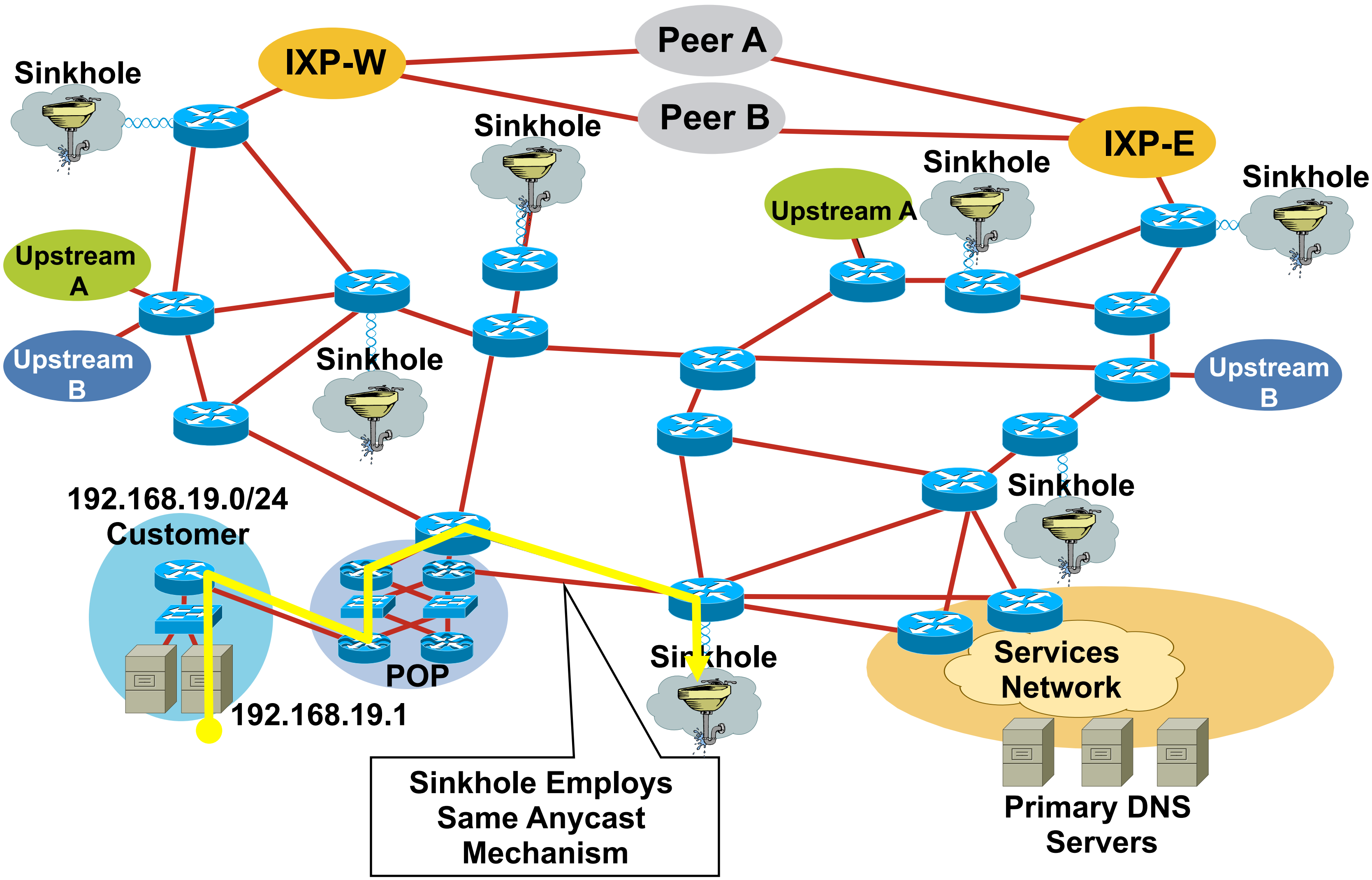
Why Sinkholes?

- They work; providers, enterprise operators and researchers use them in their network for data collection and analysis
- More uses are being found through experience and individual innovation
- Deploying sinkholes correctly takes preparation

Anycast and Sinkholes

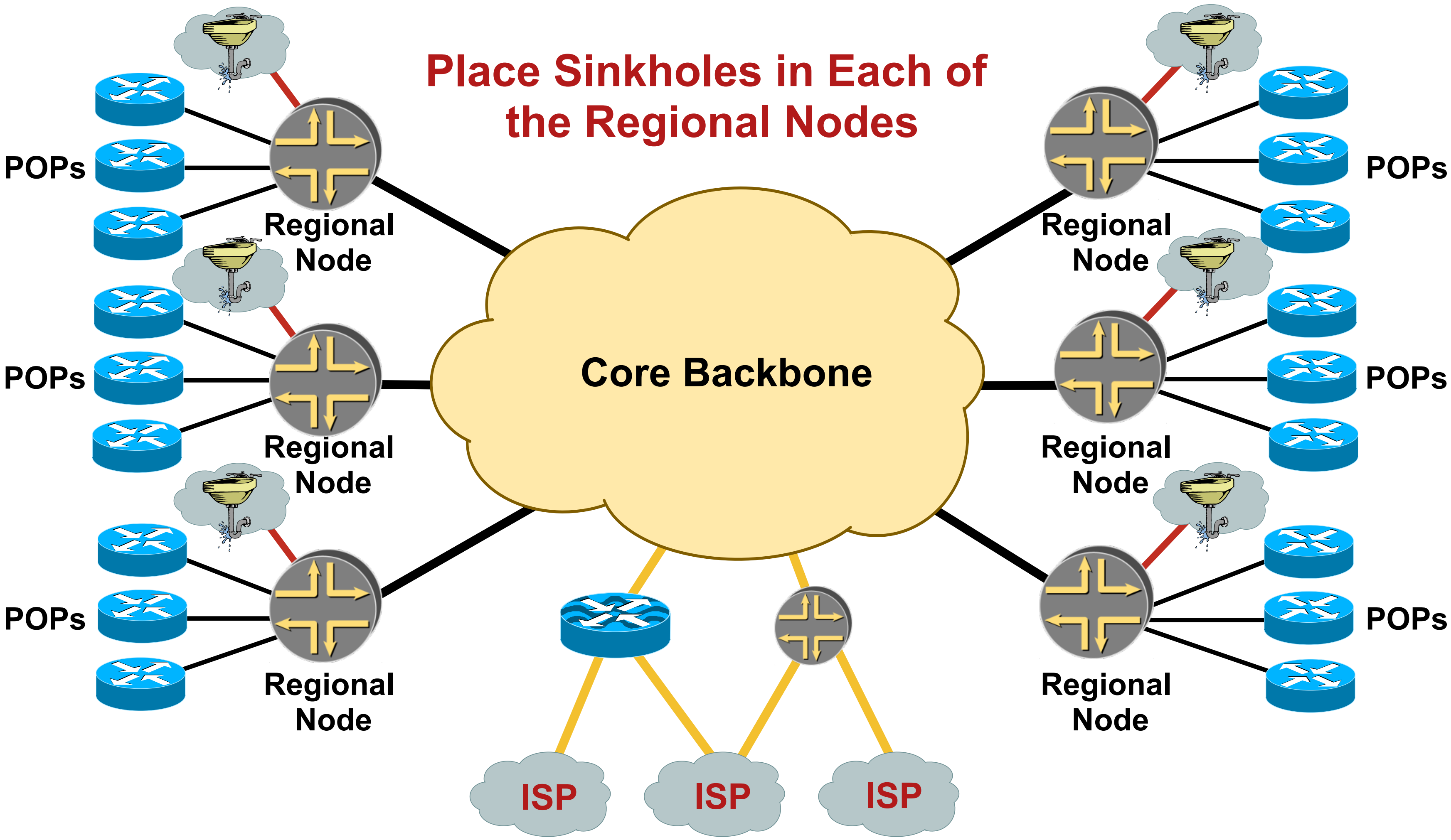
- Sinkholes are designed to pull in traffic, potentially large volumes
- Optimal placement in the network requires mindful integration and can have substantial impact on network performance and availability
- A single sinkhole might require major re-engineering of the network
- Anycast sinkholes provide a means to distribute the load throughout the network

Anycast Sinkholes

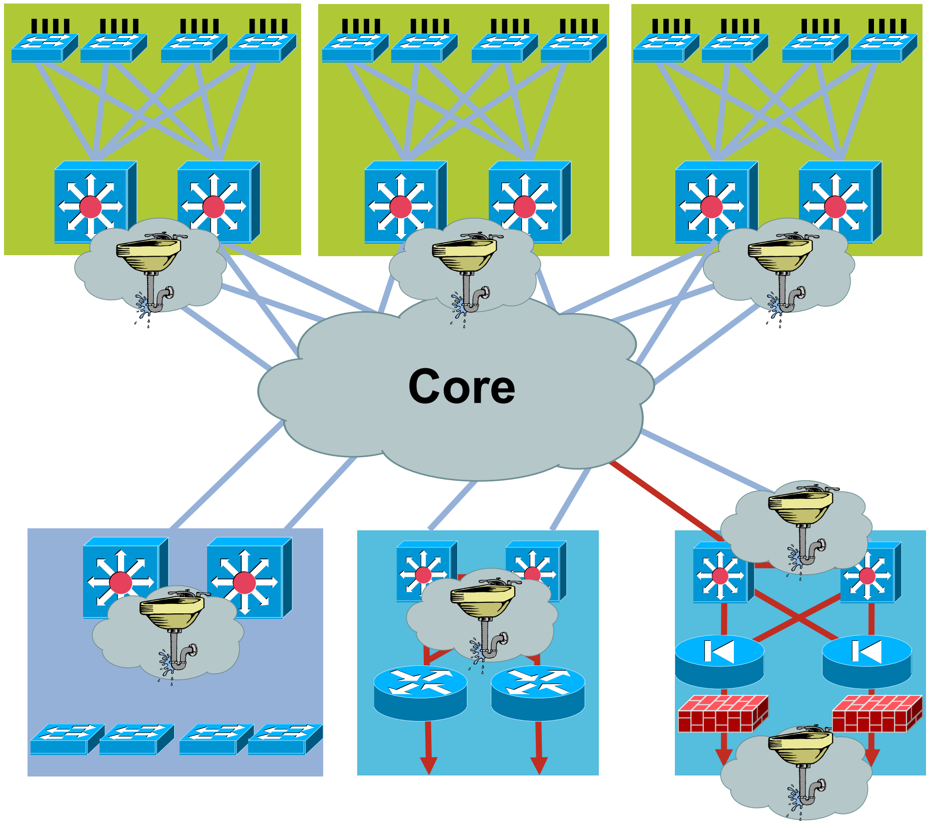


Sinkhole Employs Same Anycast Mechanism

Anycast Sinkhole Placement



Enterprise Sinkhole Placement



Server Farm

WAN

Internet

- Baselining is the key

  Measure derivations from “normal”

- Distribute sinkholes as appropriate for traffic engineering and routing architecture

- Some key locations:

  Inside internet connection

  In front of servers

  Distribution layer

Safety Precautions

- Do not allow advertisements to leak:
  - BGP no-export, no-advertise, additive communities
  - Explicit egress prefix policies (community, prefix, etc.)
- Do not allow traffic to escape the sinkhole:
  - Backscatter from a sinkhole defeats the function of a sinkhole (egress ACL on the sinkhole router)
- Advanced sinkhole designs
  - True honeypot potential → protect resources in the sinkhole
  - Don’t become part of the attack
  - Filter/rate limit outgoing connections

Reacting to Attacks

Reaction Tools

- Wide range of response options exists

  Access-control lists

  QoS tools such as CAR, traffic policing and NBAR

  Firewalls

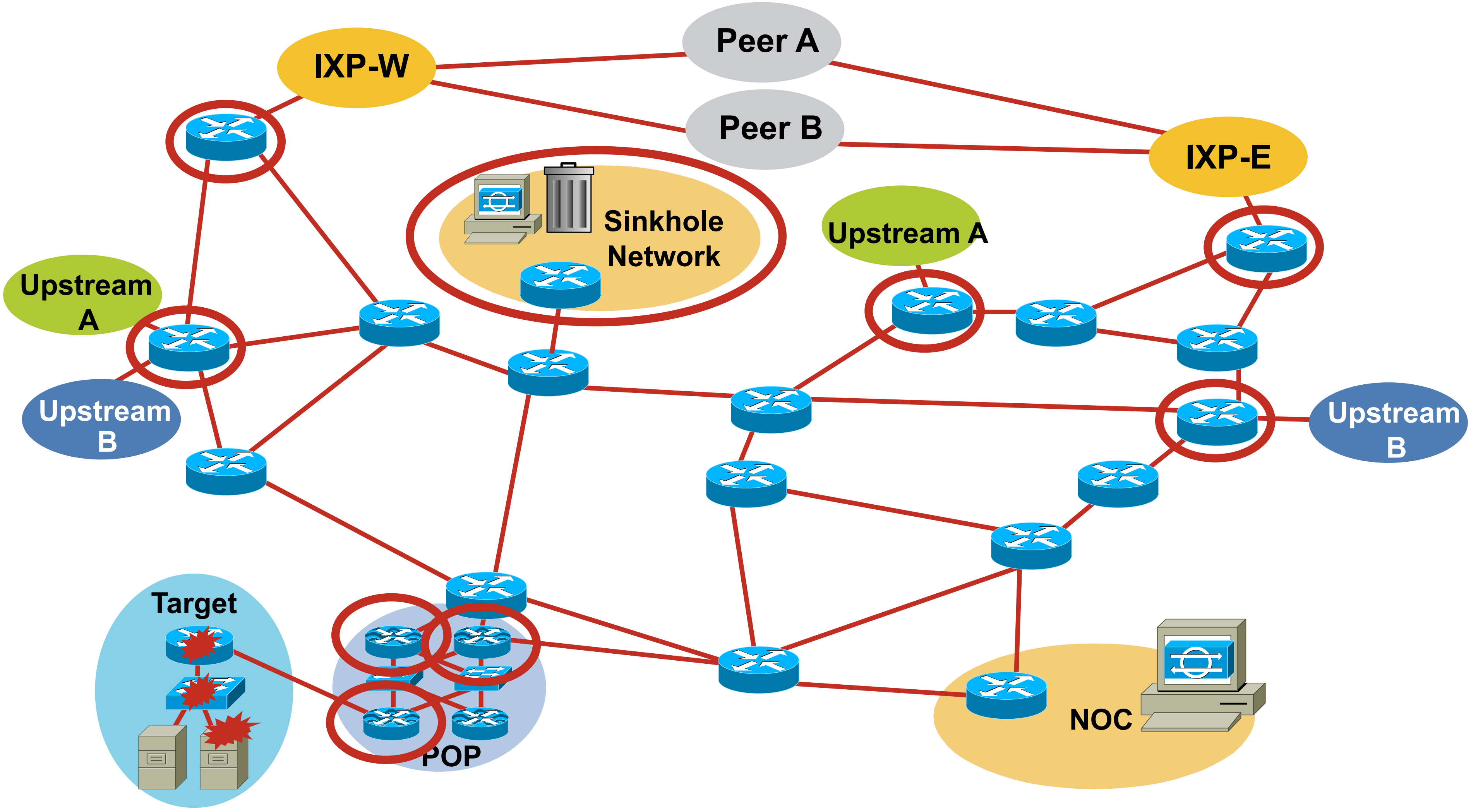
  Various IPS technologies: NIDS, HIDS, anomaly detection

  BGP triggers

  Packet scrubbing

- Today, we will focus on core-centric tools

Where to React?



QoS at the Edge as Attack Mitigation

- Tag all ingress packets at the internet edge
- Doesn’t require application or ip address awareness
- Provides proactive and reactive mitigation:
  - Proactively
    - Knocks down ToS 5-7
    - Can be added to CoPP ACL’s:

```
access-list 152 permit tcp any any eq 22 dscp af13
```

  - Reactively
    - ACL’s on the fly at internal chokepoints
    - Scavenger QoS, see:
    - Scavenger-Class QoS Strategy for DoS/Worm Attack Mitigation
    - http://www.cisco.com/application/pdf/en/us/guest/tech/tk759/c1482/cdccont_0900aecd80295ac7.pdf

QoS at the Edge as Attack Mitigation

- Configuration

```
class-map match-all edge-color
 match any
policy-map edge-color
 class edge-color
  set dscp af13

interface GigabitEthernet0/1
 service-policy input edge-color
```

- Considerations
  - CPU impact - 3825 at 50,000 pps
    - Without tagging 12% CPU
    - With tagging 25% CPU
  - Integration with existing QoS policy
  - Treats all inbound traffic equally
    - Differentiate responses to inside connections?
    - Business critical inbound connections?
    - Recolor ToS 6/7 instead?

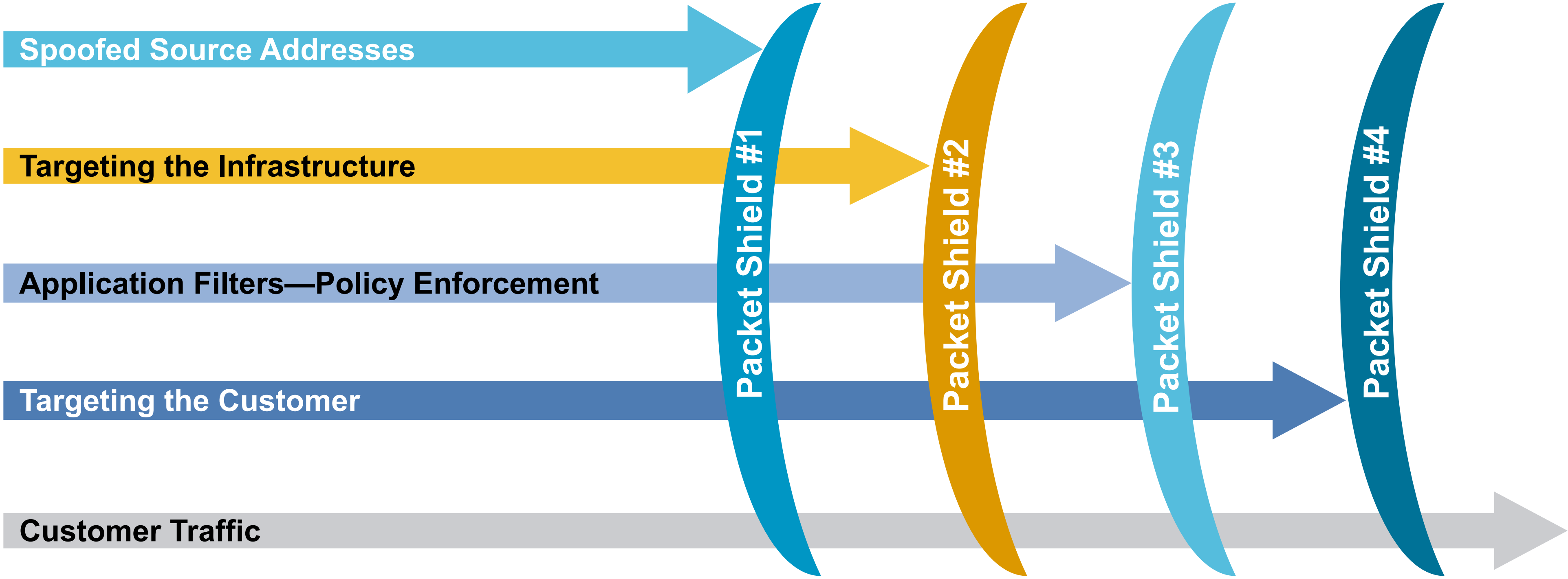
Reacting with ACLs

Reacting to an Attack with ACLs

- Traditional method for stopping attacks
- Scaling issues encountered:
  - Operational difficulties
    - Changes on the fly
    - Multiple ACLs per interface
  - Performance concerns
- How does the ACL load into the router? Does it interrupt packet flow?
- How many ACEs can be supported in hardware? In software?
- How does ACL depth impact performance?
- How do multiple concurrent features affect performance?

Packet Filtering

Viewed Horizontally



Spoofed Source Addresses

Targeting the Infrastructure

Application Filters—Policy Enforcement

Targeting the Customer

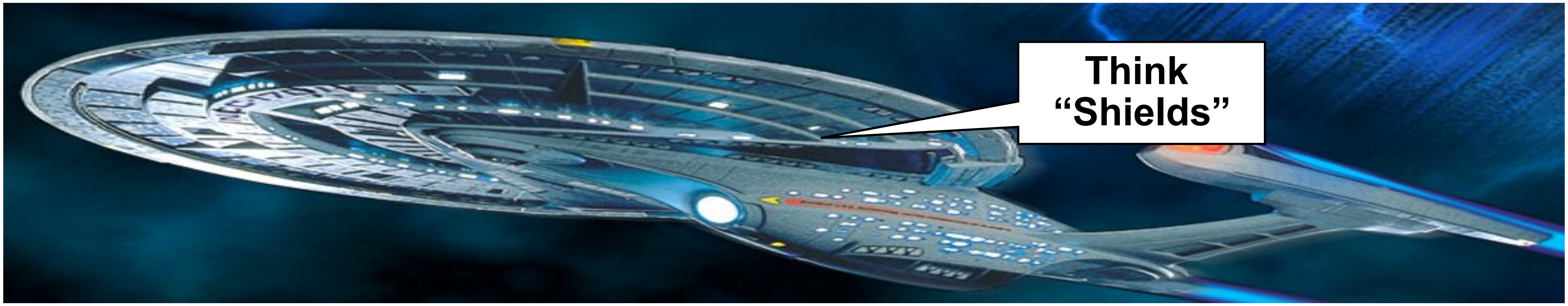
Customer Traffic

Packet Shield #1

Packet Shield #2

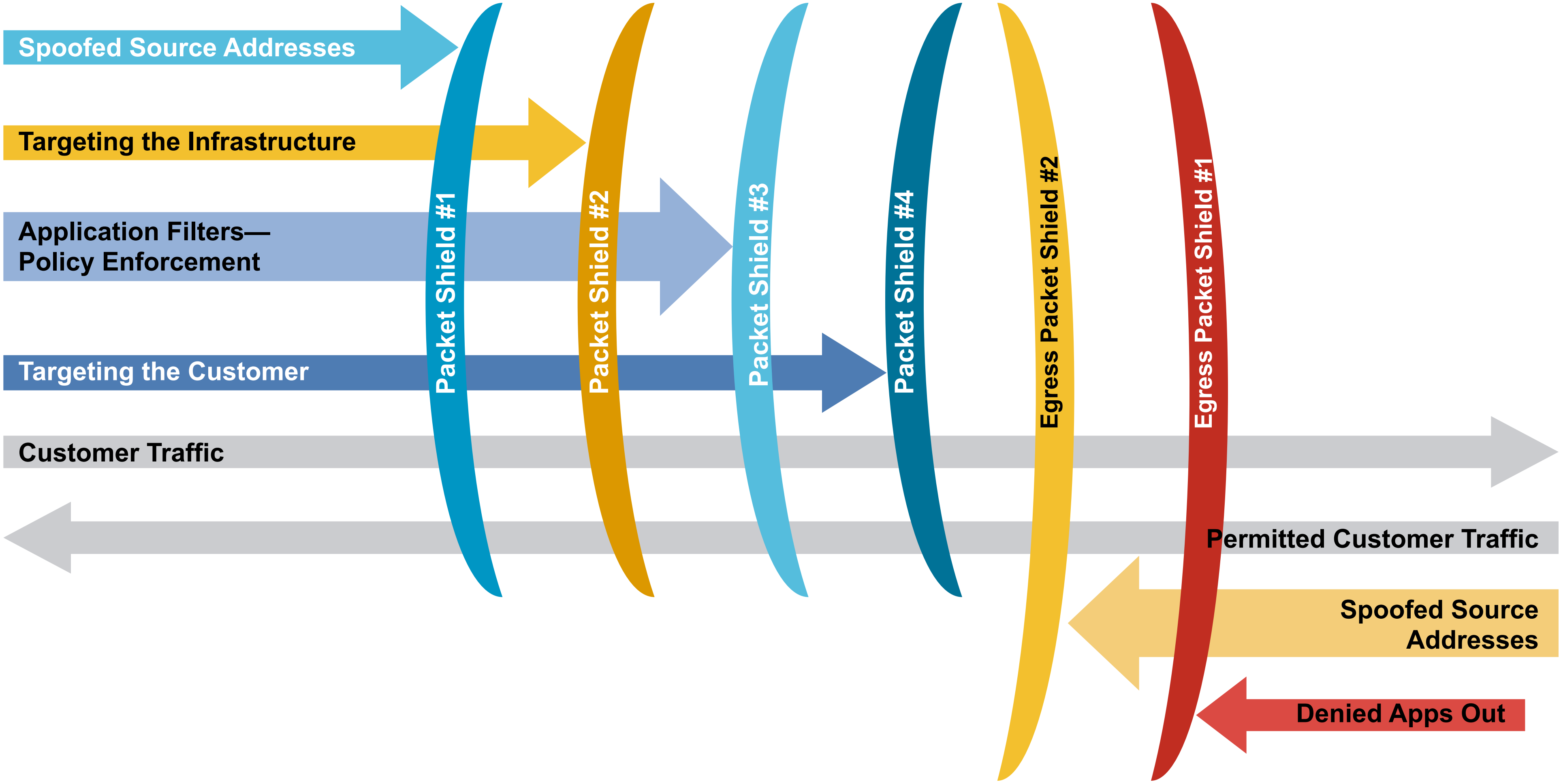
Packet Shield #3

Packet Shield #4



Packet Filtering

Remember to Filter the Return Path



ACL Summary

- ACLs are widely deployed as a primary containment tool
- Prerequisites: identification and classification—need to know what to filter
- Apply as specific an ACL as possible
- ACLs are good for static attacks, not as effective for rapidly changing attack profiles
- Understand ACL performance limitations before an attack occurs

Reacting with BGP

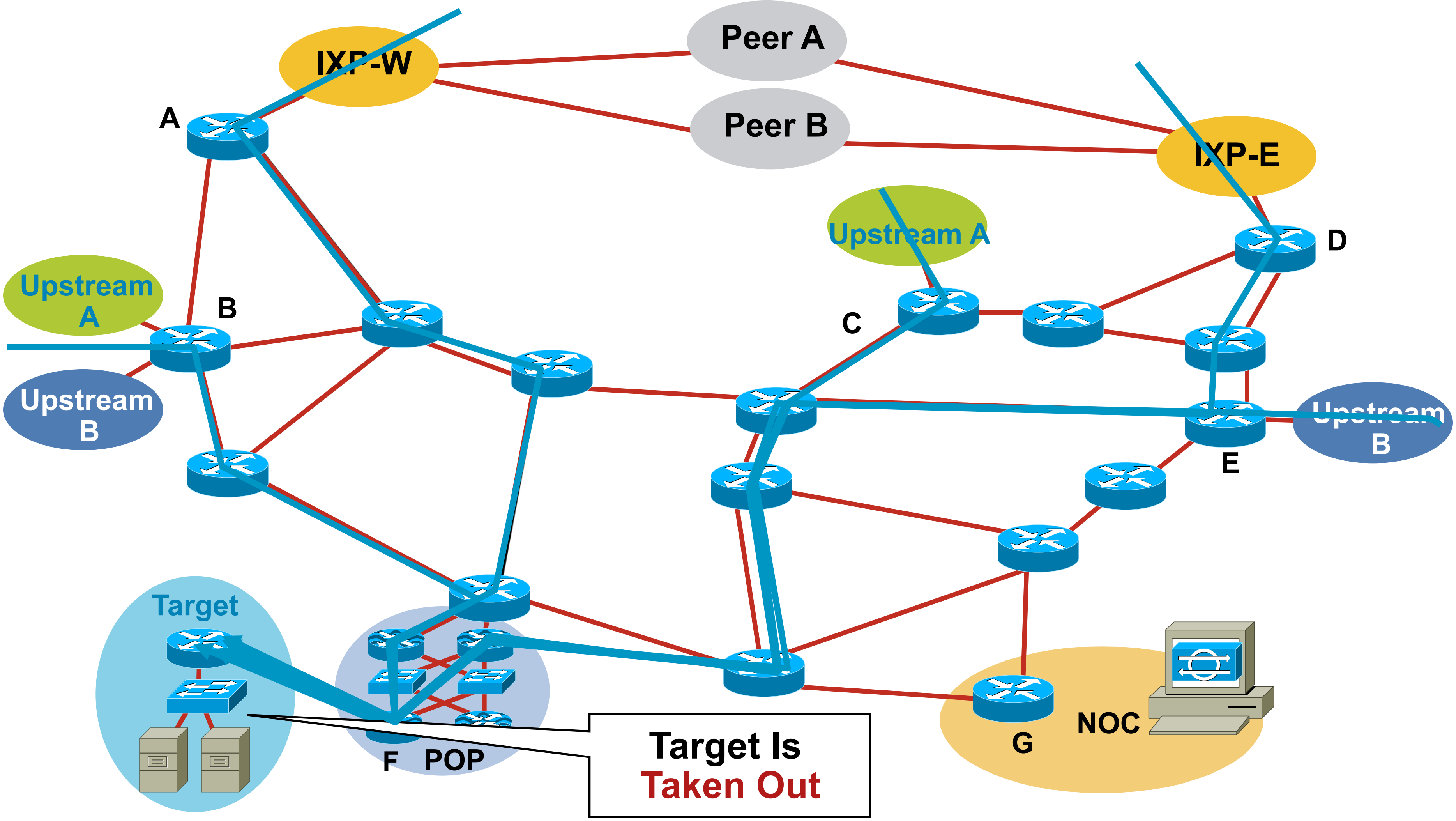
Blackhole Filtering

- Blackhole Filtering or Blackhole Routing forwards a packet to a router's bit bucket

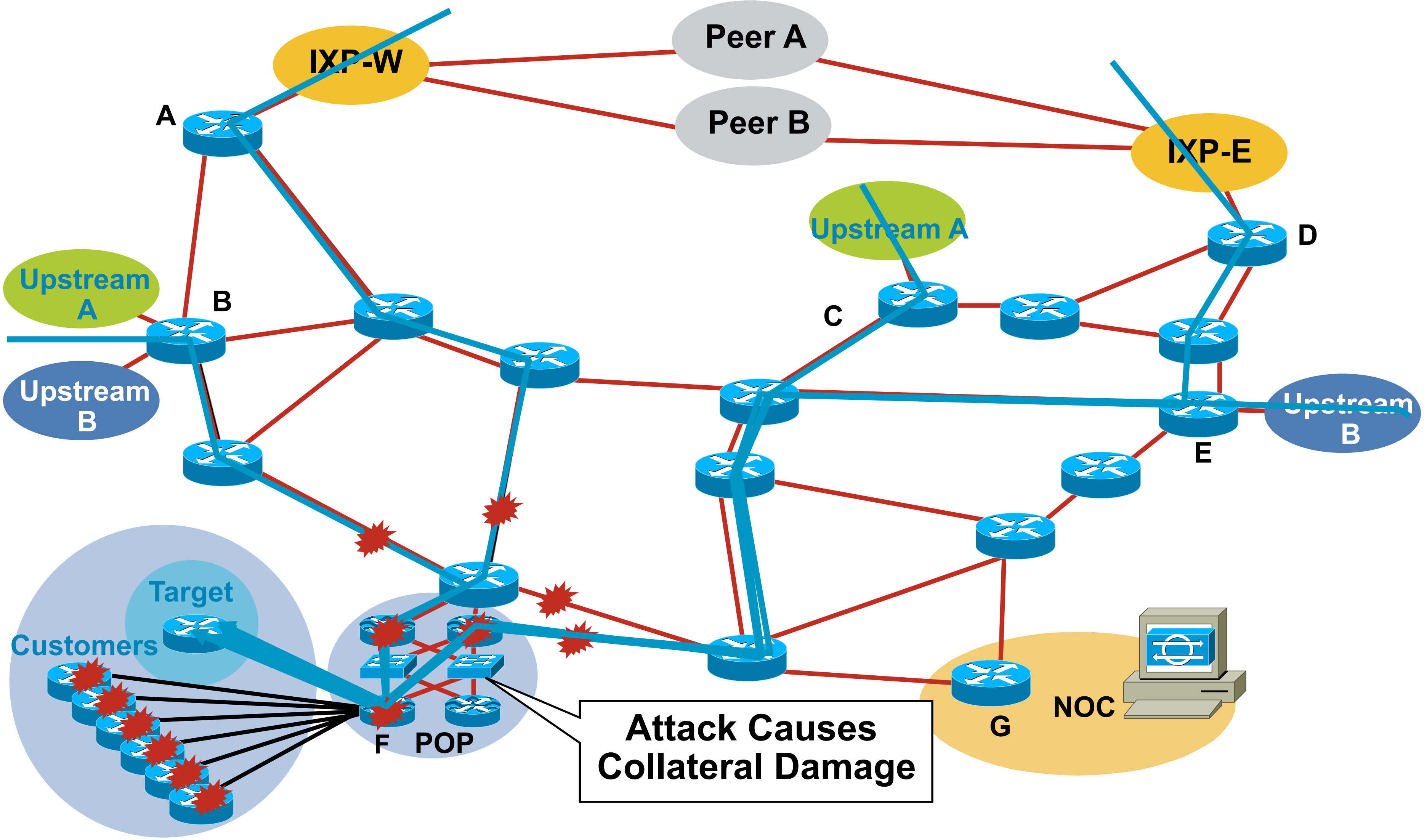
  Also known as "route to Null0"

- Works only on destination addresses, since it is really part of the forwarding logic
- Forwarding ASICs are designed to work with routes to Null0—dropping the packet with minimal to no performance impact
- Used for years as a means to "blackhole" unwanted packets

Customer Is DoSed: Before



Customer Is DoSed: Before—Collateral Damage



Attack Causes Collateral Damage

Remotely Triggered Blackhole Filtering

- We will use BGP to trigger a networkwide response to an attack
- A simple static route and BGP will enable a networkwide destination address blackhole as fast as iBGP can update the network
- This provides a tool that can be used to respond to security related events and forms a foundation for other remote triggered uses
- Often referred to as RTBH

Remote Triggered Blackhole

- Configure all edge routers with static route to Null0 (must use “reserved” network)

```
ip route 192.0.2.1 255.255.255.255 Null0
```

- Configure trigger router

  Part of iBGP mesh

  Dedicated router recommended

- Activate blackhole

  Redistribute host route for victim into BGP with next-hop set to 192.0.2.1

  Route is propagated using BGP to all BGP speaker and installed on routers with 192.0.2.1 route

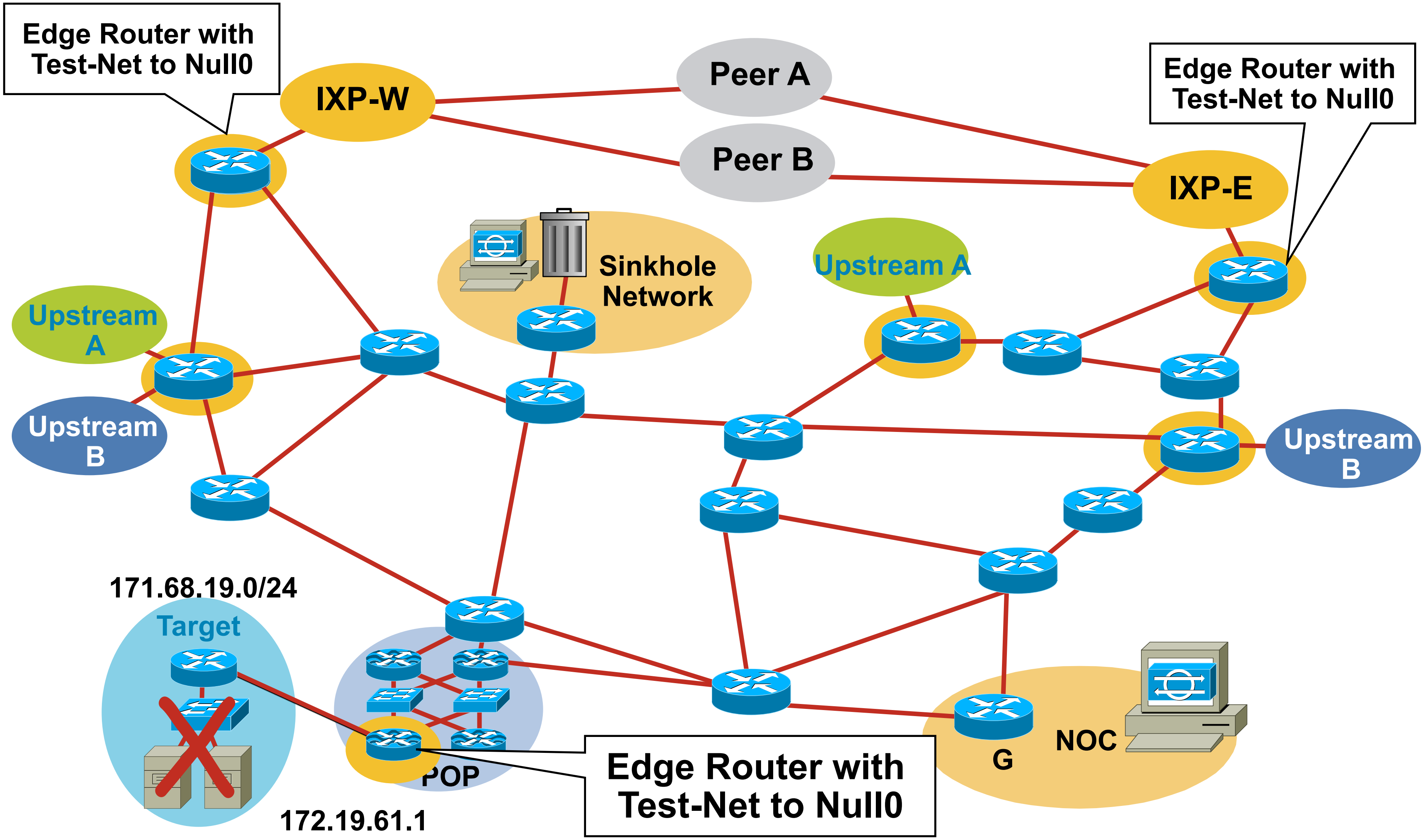
  All traffic to victim now sent to Null0

Step 1: Prepare All the Routers With Trigger

- Select a small block that will not be used for anything other than blackhole filtering; test Net (192.0.2.0/24) is optimal since it should not be in use
- Put a static route with a /32 from Test-Net—192.0.2.0/24 to Null 0 on every edge router on the network

```
ip route 192.0.2.1 255.255.255.255 Null0
```

Step 1: Prepare All the Routers With Trigger



Step 2: Prepare the Trigger Router

The Trigger Router Is the Device that Will Inject the iBGP Announcement into the ISP's Network

- Should be part of the iBGP mesh—but does not have to accept routes
- Can be a separate router (recommended)
- Can be a production router
- Can be a workstation with Zebra/Quagga (interface with Perl scripts and other tools)

Trigger Router’s Configuration

Redistribute Static with a Route-Map

```
router bgp 65535
.
redistribute static route-map static-to-bgp
.
!
route-map static-to-bgp permit 10
match tag 66
set ip next-hop 192.0.2.1
set local-preference 200
set community no-export
set origin igp
!
Route-map static-to-bgp permit 20
```

Match Static Route Tag

Set Next-Hop to the Trigger

Set Local-Pref

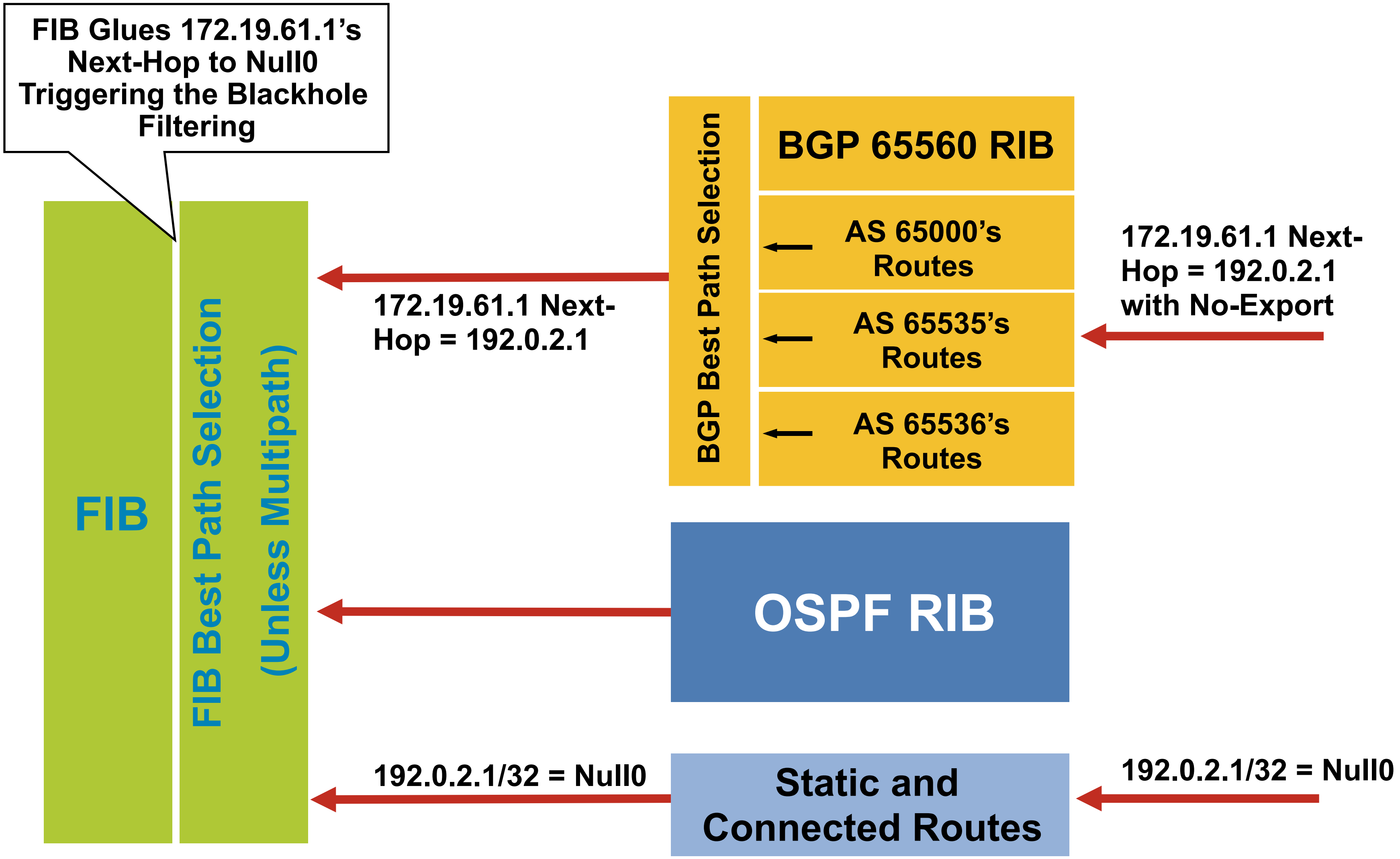
Step 3: Activate the Blackhole

- Add a static route to the destination to be blackholed; the static is added with the “tag 66” to keep it separate from other statics on the router

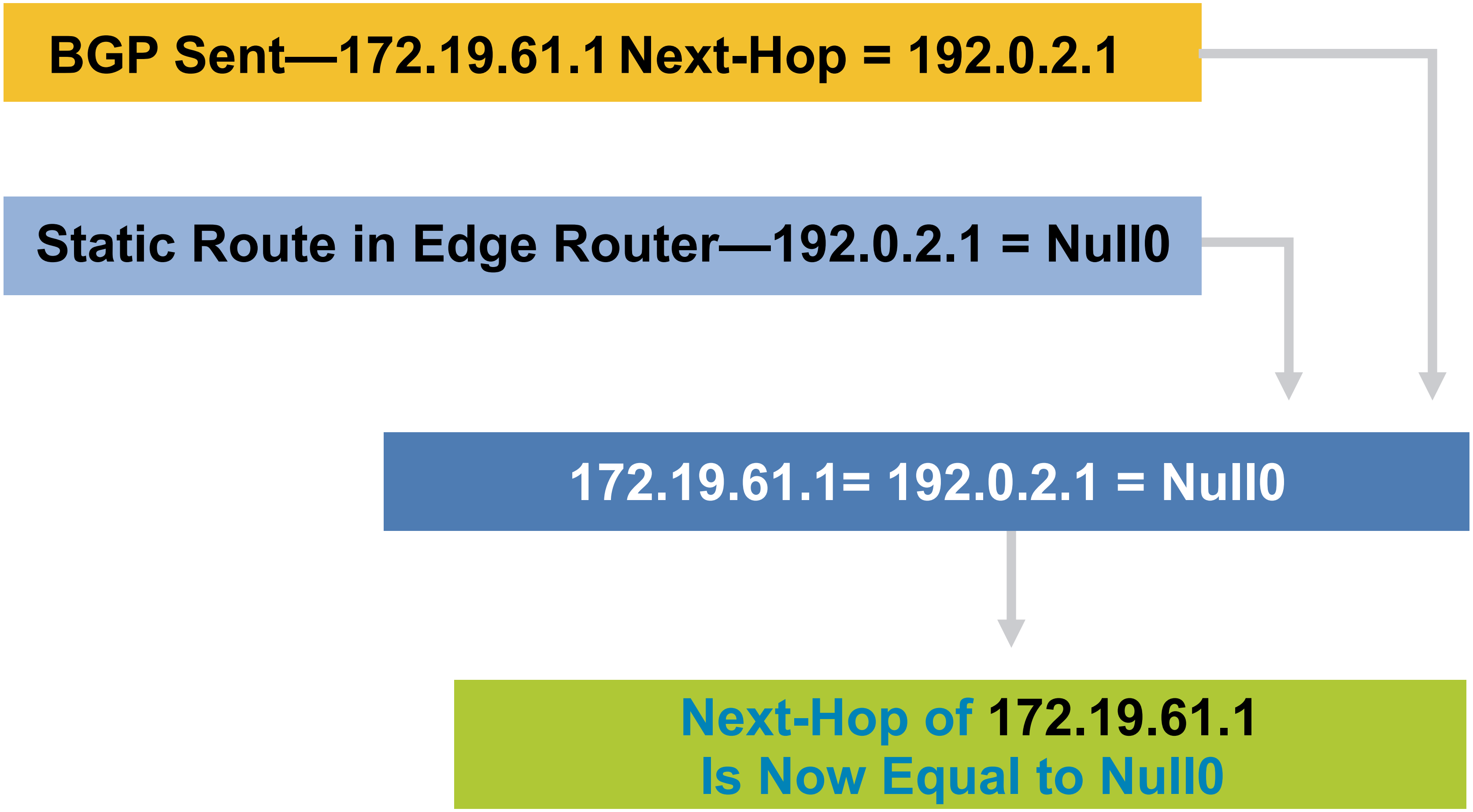
```
ip route 172.19.61.1 255.255.255.255 Null0 Tag 66
```

- BGP advertisement goes out to all BGP speaking routers
- Routers received BGP update, and “glue” it to the existing static route; due to recursion, the next-hop is now Null0

Step 3: Activate the Blackhole


FIB Glues 172.19.61.1's Next-Hop to Null0 Triggering the Blackhole Filtering
FIB
FIB Best Path Selection
(Unless Multipath)
172.19.61.1 Next-Hop = 192.0.2.1
BGP Best Path Selection
BGP 65560 RIB
AS 65000's Routes
AS 65535's Routes
AS 65536's Routes
172.19.61.1 Next-Hop = 192.0.2.1 with No-Export
OSPF RIB
192.0.2.1/32 = Null0
Static and Connected Routes
192.0.2.1/32 = Null0

Step 3: Activate the Blackhole



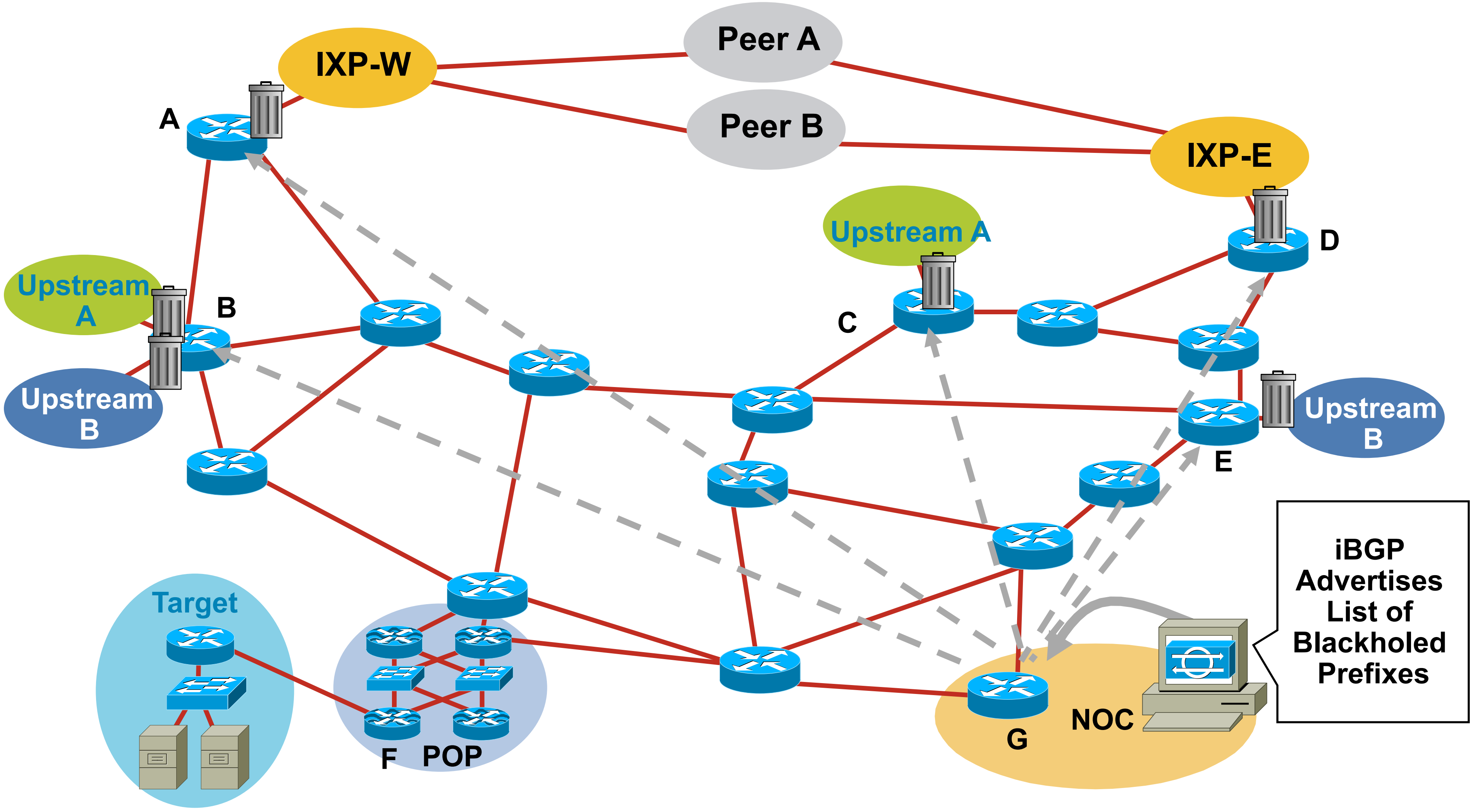
BGP Sent—172.19.61.1 Next-Hop = 192.0.2.1

Static Route in Edge Router—192.0.2.1 = Null0

172.19.61.1= 192.0.2.1 = Null0

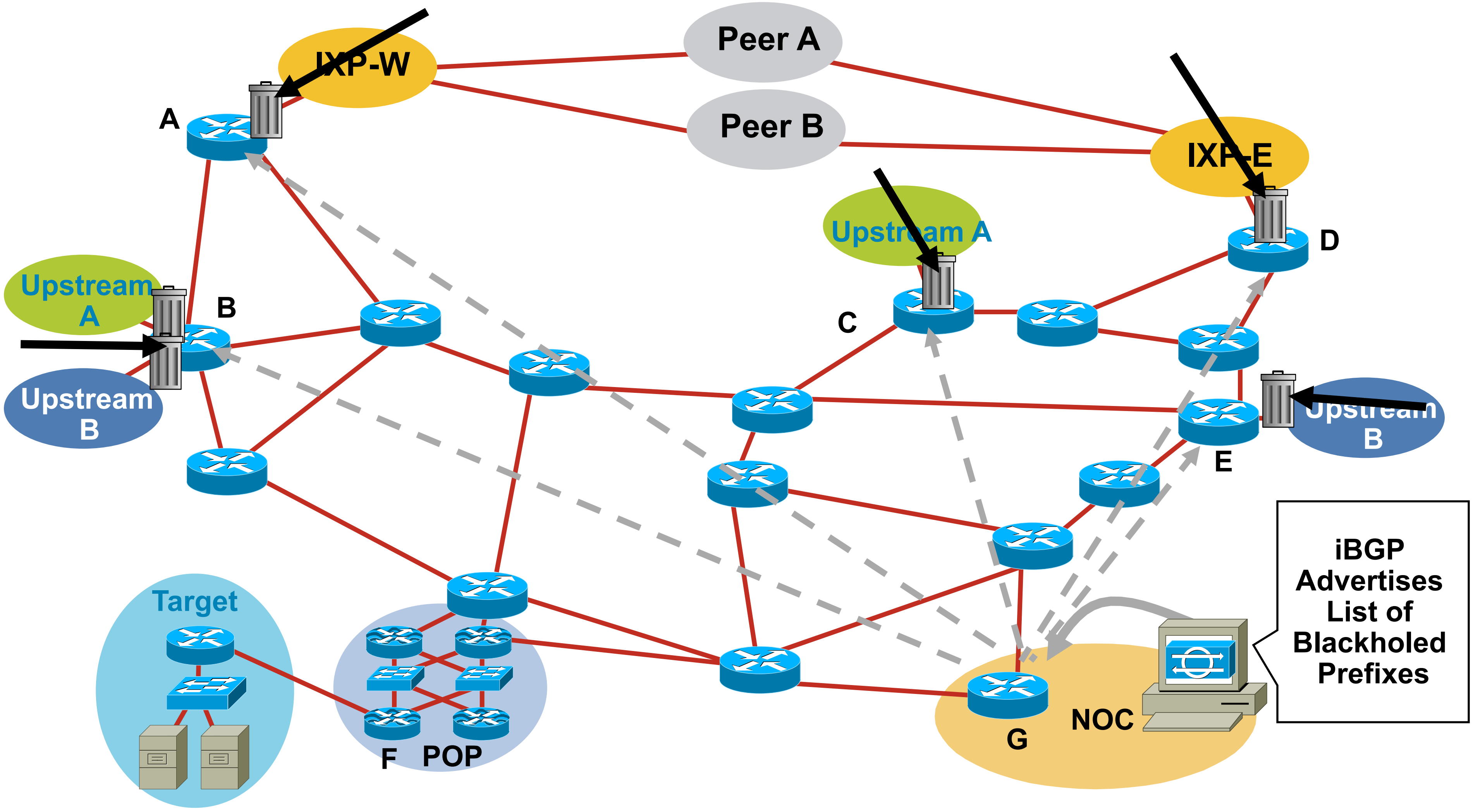
Next-Hop of 172.19.61.1
Is Now Equal to Null0

Step 3: Activate the Blackhole



iBGP Advertises List of Blackholed Prefixes

Customer Is DoSed: After—
Packet Drops Pushed to the Edge



iBGP Advertises List of Blackholed Prefixes

Using Remote Triggered Blackhole

- Is this done today?

  Yes, service providers and enterprises use frequently

- Often only scaleable answer to large-scale DoS attack

  Has proven very effective

- Interprovider triggers not implemented

  Rely on informal channels

- Service: customer triggered

  Edge customers trigger the update, SP doesn't get involved

  Implication: you detect, you classify, etc.

- White list allowed traffic to prevent self-DoS

  http://www.cymru.com/gillsr/documents/golden-networks

BGP Sinkhole Trigger

- Leverage the same BGP technique used for RTBH
- Dedicated trigger router redistributes more specific route for destination being re-rerouted
  - Next-hop set via route-map
- All BGP-speaking routers receive update
- Complex design can use multiple route-maps and next-hops to provide very flexible designs
- May require BGP on all routers

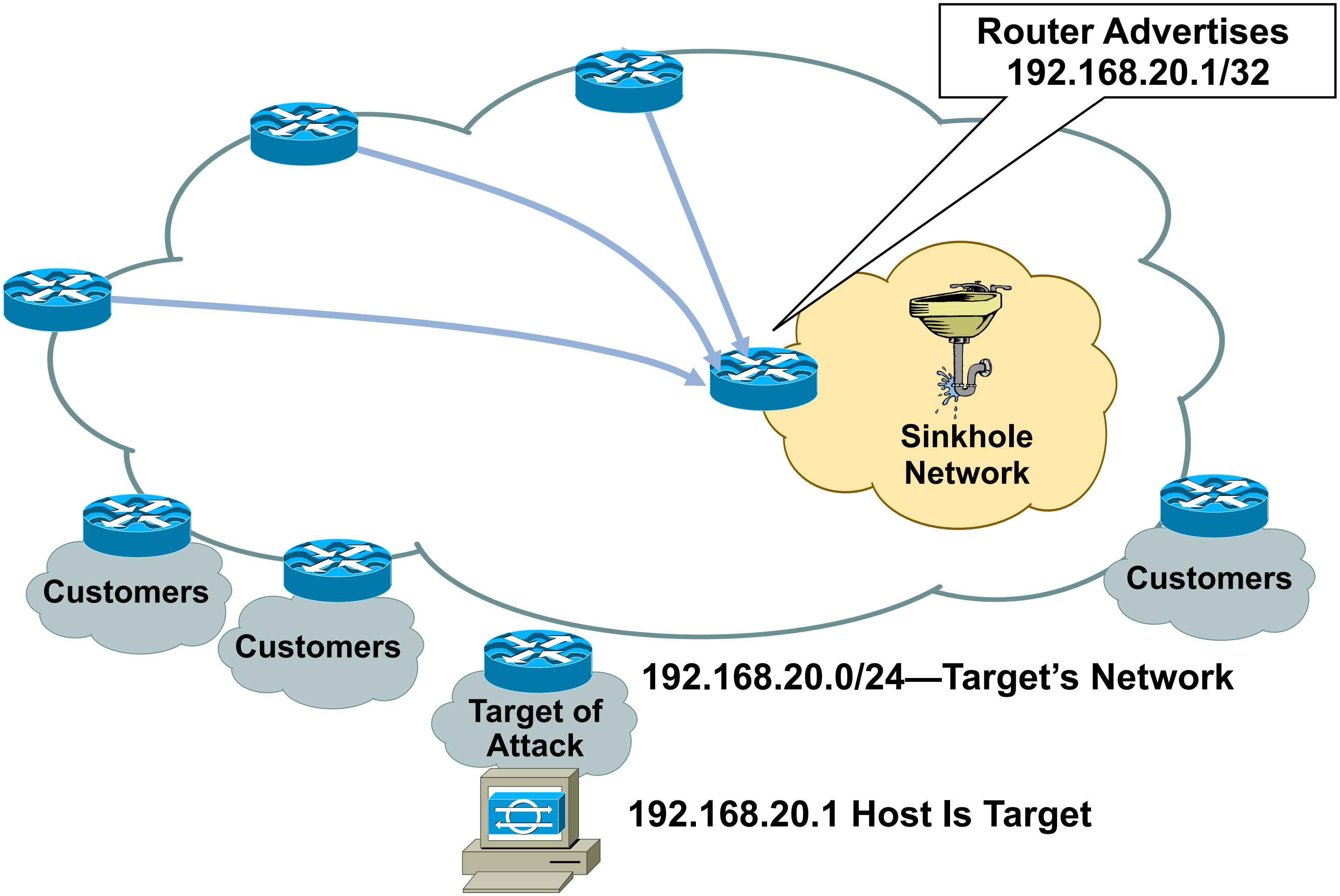
Example: BGP Sinkhole Triggers

- Sinkhole IP: 192.0.2.8
- Victim IP: 192.168.20.1
- Trigger router configuration

```
router bgp 100
redistribute static route-map static-to-bgp

route-map static-to-bgp permit 10
 match tag 66
 set origin igp
 set next-hop 192.0.2.8  <-- sinkhole address, not Null0
 set community NO-EXPORT

ip route 192.168.20.1 255.255.255.255 Null0 tag 66
```

- All traffic destined to 192.168.20.1 will be redirected to the sinkhole

Sinkhole Routers/Networks



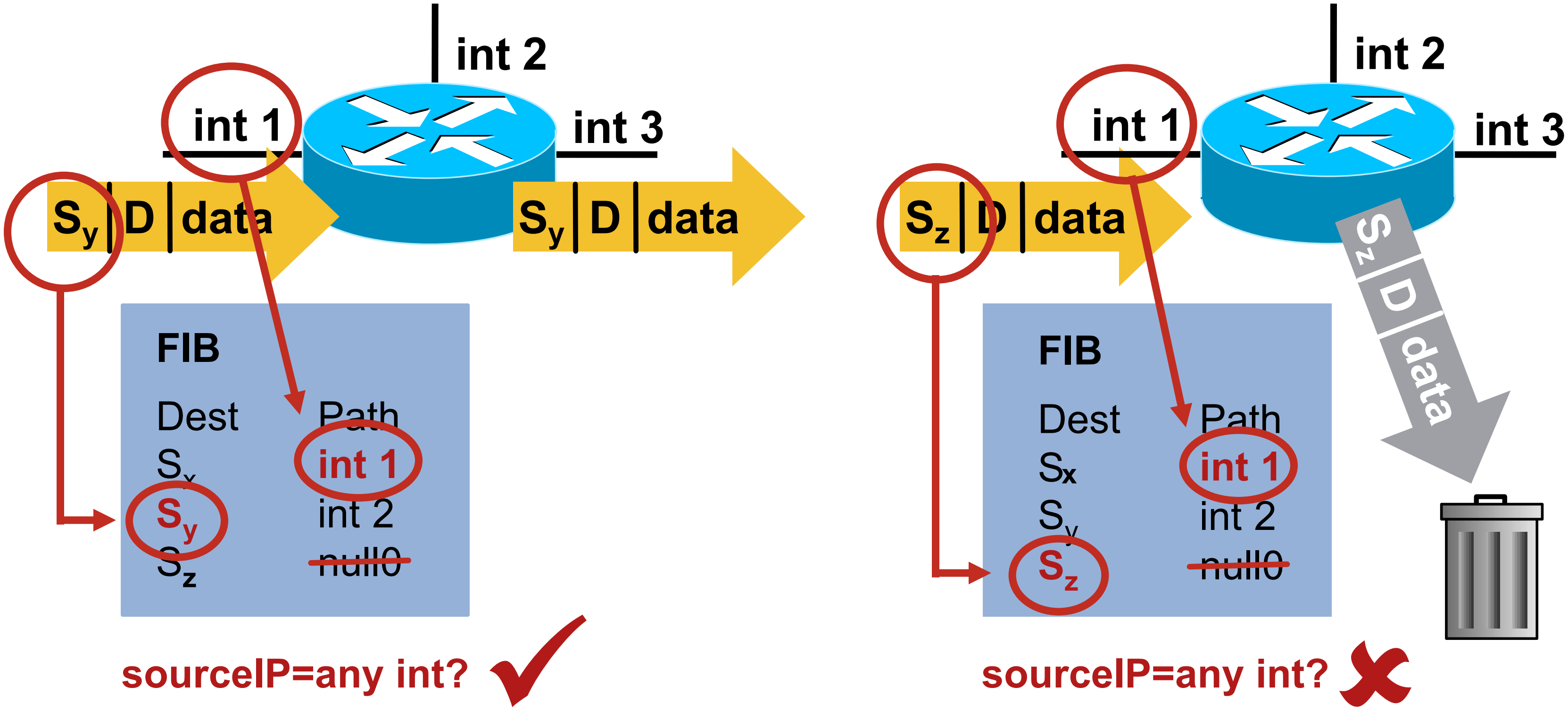
Flipping RTBH Around

Triggered Source Drops

- Dropping on destination is very important
  - Dropping on source is often what we really need
- Reacting using source address provides some interesting options:
  - Stop the attack without taking the destination offline
  - Filter command and control servers
  - Filter (contain) infected end stations
- Must be rapid and scalable
  - Leverage pervasive BGP again

Quick Review: uRPF—Loose Mode

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



FIB

Dest	Path
S_x	int 1
S_y	int 2
S_z	~~null0~~

FIB

Dest	Path
S_x	int 1
S_y	int 2
S_z	~~null0~~

IP Verify Unicast Source Reachable—Via any

Source-Based Remote Triggered

Blackhole Filtering

Uses the Same Architecture as Destination-Based Filtering + Unicast RPF

- Edge routers must have static in place
- They also require Unicast RPF
- BGP trigger sets next hop—in this case the “victim” is the source we want to drop

Source-Based Remote Triggered

Blackhole Filtering

- What do we have?

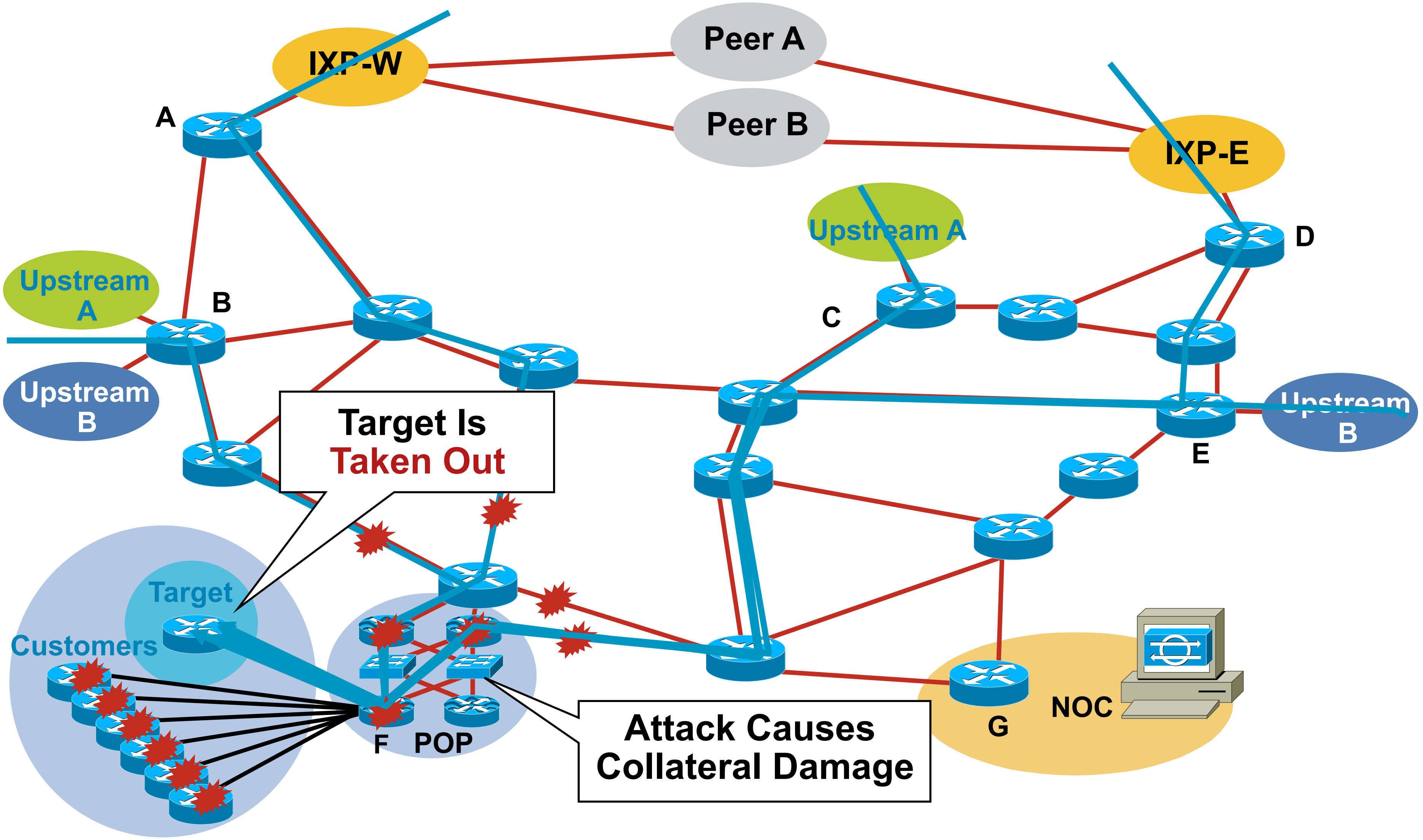
  Blackhole Filtering—if the destination address equals Null0, we drop the packet

  Remote Triggered—trigger a prefix to equal Null0 on routers across the Network at iBGP speeds

  uRPF Loose Check—if the source address equals Null0, we drop the packet

- Put them together and we have a tool to trigger drop for any packet coming into the network whose source or destination equals Null0

Customer Is DoSed: Before

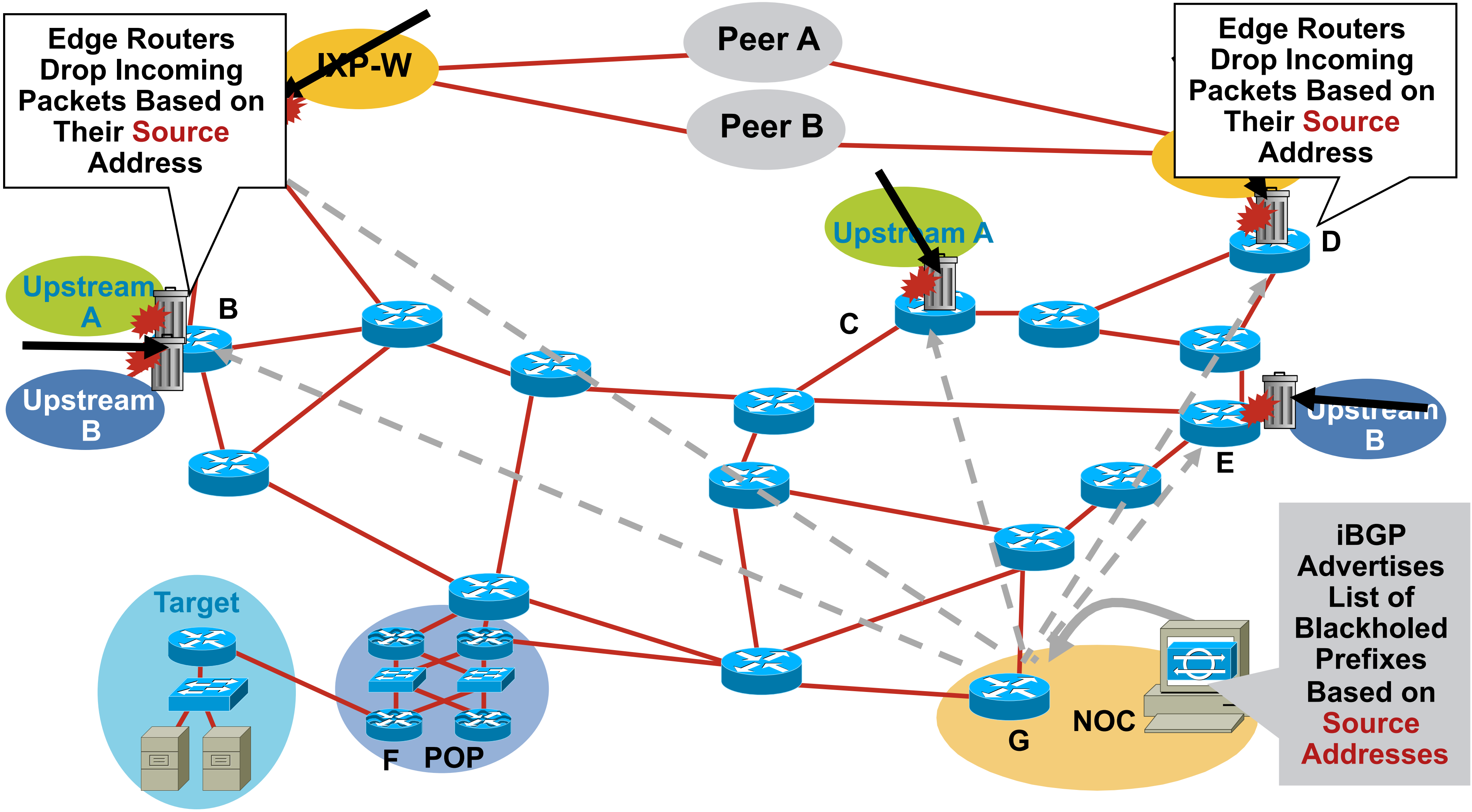


Target Is Taken Out

Attack Causes Collateral Damage

Customer Is DoSed: After

Packet Drops Pushed to the Edge



Edge Routers Drop Incoming Packets Based on Their Source Address

Edge Routers Drop Incoming Packets Based on Their Source Address

iBGP Advertises List of Blackholed Prefixes Based on Source Addresses

Possible Remote Trigger Placement

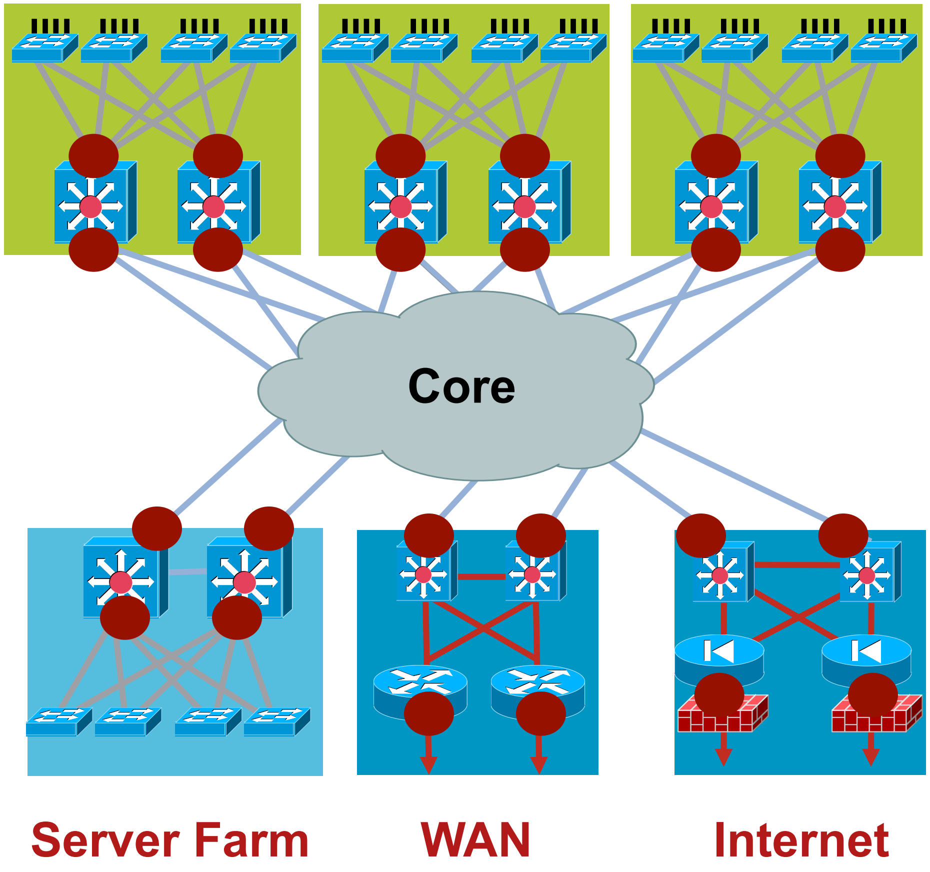
- Dark red dots indicate possible remote drop location
- L3 boundaries between network components

  Drop infections at distribution layer

  Drop incoming Internet attack at Internet edge

  React to incoming attacks from remote office across the WAN

  etc.



Community-Based Trigger

- BGP community-based triggering allow for more fined tuned control over where you drop the packets
- Three parts to the trigger:

  Static routes to Null0 on all the routers

  Trigger router sets the community

  Reaction routers (on the edge) matches community and sets the next-hop to the static route to Null0

Why Community-Based Triggering?

Allows for More Control on the Attack Reaction

- Trigger community #1 can be for all routers in the network
- Trigger community #2 can be for all peering routers; no customer routers - allows for customers to talk to the DoSed customer within your AS
- Trigger community #3 can be for all customers; used to push a inter-AS traceback to the edge of your network
- Trigger communities per ISP Peer can be used to only blackhole on one ISP Peer’s connection; allows for the DoSed customer to have partial service

BGP: Not Just For Routing, Anymore

- “I don’t want to use BGP as a routing protocol”
  - Think of BGP as a signaling protocol
  - Routing protocols operate as “ships in the night”
- BGP has a unique property among routing protocols: arbitrary next hops can be administratively defined
- There is no need to actually carry routes in BGP
  - Deploy iBGP mesh internally and do not use it for routing
  - Under normal conditions, BGP holds zero routes
  - When used for drops, only the blackholed addresses are in the table
- If BGP is used for inter-region routing, drop boundaries can be both local within a campus and global
  - Use communities to “scope” the drops

Internal Source-Based Drops

- Both source and destination drops can be used internally
  - Source drops likely the most interesting case
  - Destination drops still result in target DoS
  - Don’t forget the Internet and WAN edges
- Provides a very effective mechanism to handle internal attacks
  - Drop worm infected PCs off the network
  - Drop “owned” devices off the network
  - Protect the infrastructure
  - Whitelist to prevent self DoS

Source-Based RTBH

Key Advantages

- No ACL update
- No change to the router's configuration
- Drops happen in the forwarding path
- Frequent changes when attacks are dynamic (for multiple attacks on multiple customers)

What If I Can’t Deploy RTBH?

- Start with uRPF and static routes to NULL0
- Results in traffic source drops

```
interface g0/0
 ip verify unicast source reachable-via rx allow-default
ip route 10.0.0.0 255.0.0.0 Null0
ip route 169.254.0.0 255.255.0.0 Null0
ip route 172.16.0.0 255.240.0.0 Null0
ip route 192.0.2.0 255.255.255.0 Null0
ip route 192.168.0.0 255.255.0.0 Null0
```

- For example, traffic from 10.1.1.1 will be discarded
- Can be deployed in reaction to attacks
- A start but… won’t be fast and doesn’t scale

ACLs or uRPF Remote-Triggered Drop?

- ACLs key strengths:

  Detailed packet filtering (ports, protocols, ranges, fragments, etc.)

  Relatively static filtering environment

  Clear filtering policy

- ACLs can have issues when faced with:

  Dynamic attack profiles (different sources, different entry points, etc.)

  Frequent changes

  Quick, simultaneous deployment on a multitude of devices

- Combining ACLs with uRPF remote-triggered drops allows for ACLs to handle the strict static policies while uRPF remote-triggered blackhole handles the dynamic source-based drops

References

- DoS detection:

  "Tackling Network DoS on Transit Networks": David Harmelin, DANTE, March 2001

  http://www.dante.net/pubs/dip/42/42.html

  "Inferring Internet Denial-of-Service Activity": David Moore et al, May 2001

  http://www.caida.org/outreach/papers/2001/BackScatter/usenixsecurity01.pdf

  "The Spread of the Code Red Worm": David Moore, CAIDA, July 2001

  http://www.caida.org/analysis/security/code-red/

- DoS tracing:

  "Tracing Spoofed IP Addresses": Rob Thomas, Feb 2001
  (good technical description of using NetFlow to trace back a flow)

  http://www.cymru.com/Documents/tracking-spoofed.html

- Other:

  "DoS Attacks against GRC.com": Steve Gibson, GRC, June 2001 (a real-life description of attacks from the victim side; somewhat disputed, but fun to read)

  http://grc.com/dos/grcdos.htm

  SECURITY@CISCO

  http://www.cisco.com/security/

NetFlow—More Information

- Cisco NetFlow home

  http://www.cisco.com/en/US/tech/tk812/tsd_technology_support_protocol_home.html

- Linux NetFlow reports HOWTO

  http://www.dynamicnetworks.us/netflow/netflow-howto.html

- Arbor Networks PeakFlow SP

  http://www.arbornetworks.com/products_sp.php

SNMP—More Information

- Cisco SNMP object tracker

  http://www.cisco.com/pcgi-bin/Support/Mibbrowser/mibinfo.pl?tab=4

- Cisco MIBs and trap definitions

  http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

- SNMPLink

  http://www.snmplink.org/

RMON—More Information

- IETF RMON WG

  http://www.ietf.org/html.charters/rmonmib-charter.html

- Cisco RMON home

  http://www.cisco.com/en/US/tech/tk648/tk362/tk560/tsd_technology_support_sub-protocol_home.html

- Cisco NAM product page

  http://www.cisco.com/en/US/products/hw/modules/ps2706/ps5025/index.html

Packet Capture—More Information

- tcpdump/libpcap home

  http://www.tcpdump.org/

- Vinayak Hegde’s Linux Gazette article

  http://linuxgazette.net/issue86/vinayak.html

Syslog—More Information

- Syslog.org

  http://www.syslog.org/

- Syslog logging with PostGres HOWTO

  http://kdough.net/projects/howto/syslog_postgresql/

- Agent Smith explains Syslog

  http://routergod.com/agentsmith/

BGP—More Information

- Cisco BGP home

  http://www.cisco.com/en/US/tech/tk365/tk80/tsd_technology_support_sub-protocol_home.html

- Slammer/BGP analysis

  http://www.cs.colostate.edu/~massey/pubs/conf/massey_iwdc03.pdf

- Team CYMRU BGP tools

  http://www.cymru.com/BGP/index.html

Traceback—Direct Contact Information

- APNIC—reporting network abuse: spamming and hacking

  http://www.apnic.net/info/faq/abuse/index.html

- RIPE—reporting network abuse: spamming and hacking

  http://www.ripe.net/info/faq/abuse/index.html

- ARIN—network abuse: FAQ

  http://www.arin.net/abuse.html

References

- Product security:

  Cisco's product vulnerabilities

  http://www.cisco.com/en/US/products/products_security_advisories_listing.html

  Cisco Security Center

  http://www.cisco.com/security

- ISP essentials:

  Technical tips for ISPs every ISP should know

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

- Technical tips:

  Troubleshooting High CPU Utilization on Cisco Routers

  http://www.cisco.com/warp/public/63/highcpu.html

  The "show processes" command

  http://www.cisco.com/warp/public/63/showproc_cpu.html

  NetFlow performance white paper

  http://www.cisco.com/en/US/partner/tech/tk812/technologies_white_paper0900aecd802a0eb9.shtml

- Mailing list:

  cust-security-announce@cisco.com: all customers should be on this list

Q and A