

ISP Security

Securing the network infrastructure of an ISP and responding to attack traffic

SANOG14, Chennai, India

Gaurab Raj Upadhaya,PCH and Rehan Nedaria, Cisco

ISP Security Incident Response

- **ISP's Operations Team response to a security incident can typically be broken down into six phases:**

Preparation

Identification

Classification

Traceback

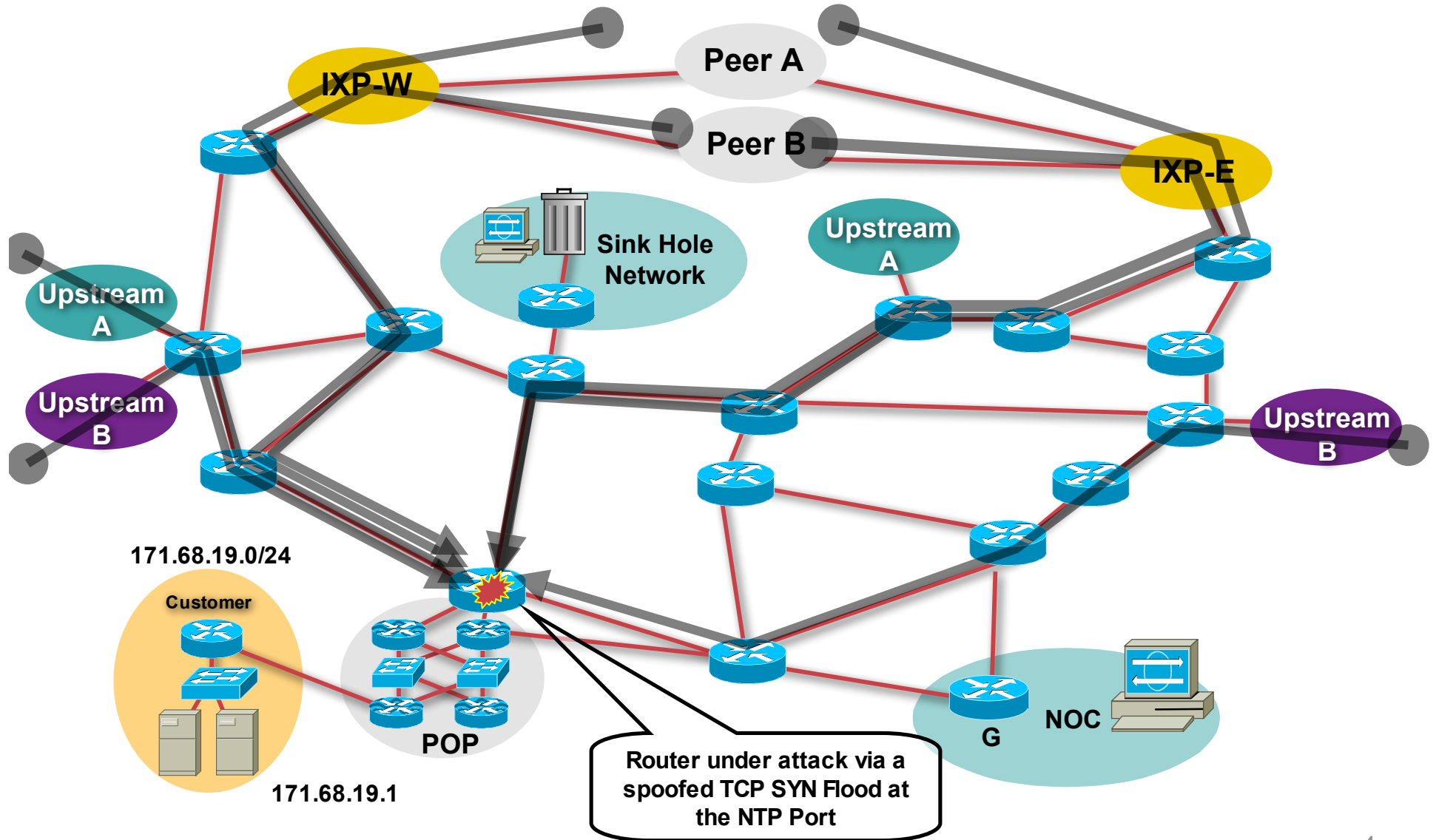
Reaction

Post Mortem

Phase 1 – Preparation for the Attack

Securing the Router and the Management Plane

Routers do get Directly Attacked



Terminology and Simple Risk Assessment

- **Three Plane Conceptual Model:**

Data Plane – Packets going through the router.

Control Plane – The routing protocols gluing the network together.

Management Plane – The tools and protocols used to manage the device.

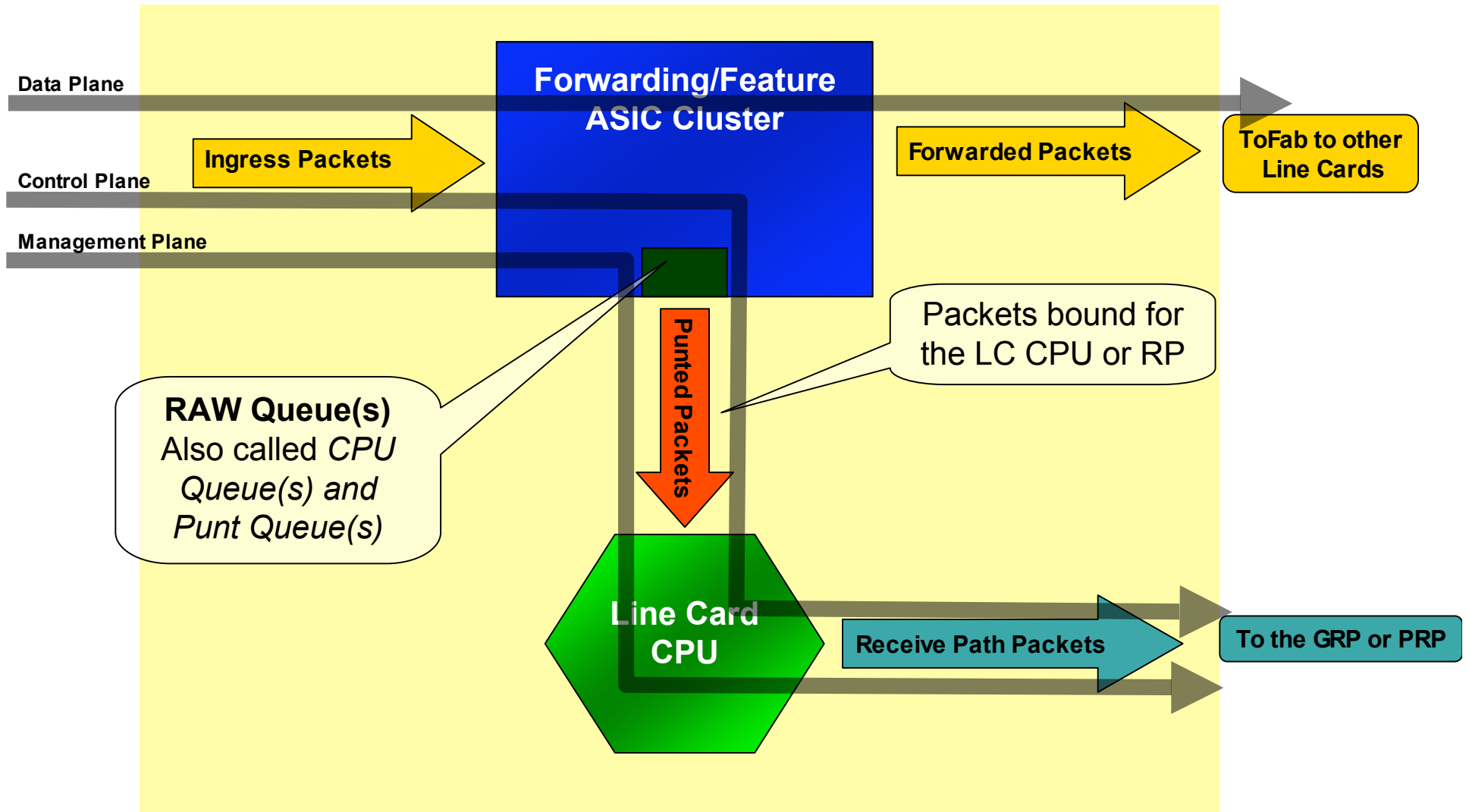
- **Direct Attacks on the Router usually hits on three attack vectors:**

Bandwidth Saturation (Data plane)

Target the Control 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).

The Three *Planes*



Securing the Router

Essential Lock Downs

Global Services You Turn OFF

- **Some services turned on by default, should be turned off to save memory and prevent security breaches/attacks**

```
n o   i p   f i n g e r
```

```
n o   s e r v i c e   p a d
```

```
n o   s e r v i c e   u d p - s m   a l l - s e r v e r s
```

```
n o   s e r v i c e   t c p - s m   a l l - s e r v e r s
```

```
n o   i p   b o o t p   s e r v e r
```

Global Services You Turn OFF

- **Finger**

Find out who is logged in, from where, how long for

- **PAD**

Historical – from the days of X.25

- **Small servers**

Tcp and udp ports < 20 are for developing IP stacks and not needed in day to day operations

- **Bootp**

Used by systems to bootstrap themselves onto the network – e.g. X-terminals

Interface Services You Turn OFF

- **Some IP features are great for campus LANs, but do not make sense on a ISP backbone**
- **All interfaces on an ISP's backbone router should have the follow as a default:**

```
n o i p r e d i r e c t s
```

```
n o i p d i r e c t e d - b r o a d c a s t
```

```
n o i p p r o x y - a r p
```

Interface Services You Turn OFF

- **IP redirects**

Router will send redirect message if it has to resend a packet through the same interface it was received on

- **Direct-broadcast**

If packet intended for network broadcast address, router will physically broadcast it onto the attached network

The cause of all SMURF attacks on the Internet

- **Proxy-arp**

Dumb host sends arp request for destination – documented in RFC1027

If router knows how to get to that destination, it will install an entry in the arp table for that destination

Cisco Discovery Protocol

- Lets network administrators discover neighbouring Cisco equipment, model numbers and software versions
- Should not be needed on ISP network

`no cdp run`

- Should not be activated on any public facing interface: IXP, customer, upstream ISP – unless part of the peering agreement.
- Disable per interface

`no cdp enable`

Cisco Discovery Protocol

```
Defiant# show cdp neighbors detail

-----

Device ID : Excalibur

Entry address(es):

  IP address: 4.1.2.1

Platform : cisco RSP2 , Capabilities : Router

Interface : FastEthernet1/1 , Port ID (outgoing port): FastEthernet4/1/0

Holdtime : 154 sec

Version :

Cisco Internetwork Operating System Software

IOS (tm) RSP Software (RSP-K3PV-M), Version 12.0(9.5)S, EARLY DEPLOYMENT MAINTEN
ANCE INTERIM SOFTWARE

Copyright (c) 1986-2000 by Cisco Systems, Inc.

Compiled Fri 03-Mar-00 19:28 by htseng

Defiant#
```

Login Banner

- **Use a good login banner, or nothing at all:**

```
banner login ^  
  Authorised access only  
  This system is the property of MattNet Internet  
  Disconnect IMMEDIATELY if you are not an authorised user!  
  Contact noc@mattnet.net +99 999 999999 for help.  
^
```

- **This is a legal requirement in some jurisdictions. Check with your legal group.**

Exec Banner

- **Useful to remind logged in staff of local 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 +99 999 999999 for more info.

^

Use Enable Secret

- Encryption '7' on a Cisco is reversible
- The “enable secret” password encrypted via a one-way algorithm

```
e n a b l e   s e c r e t   < r e m o v e d >
```

```
n o   e n a b l e   p a s s w o r d
```

```
s e r v i c e   p a s s w o r d - e n c r y p t i o n
```

Securing Access to the Router

ISP Tools to Secure Access to the Router

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



VTY and Console Port Timeouts

- **Default idle timeout on async ports is 10 minutes 0 seconds**

```
e x e c - t i m e o u t 1 0 0
```

- **Timeout of 0 means permanent connection**
- **TCP keepalives on incoming network connections**

```
s e r v i c e t c p - k e e p a l i v e s - i n
```

- **Kills unused connections**

VTY Security

- **Access to VTYs should be controlled, not left open; consoles should be used for last resort admin only:**

```
a c c e s s - l i s t 3   p e r m i t 2 1 5 . 1 7 . 1 . 0   0 . 0 . 0 . 2 5 5
```

```
a c c e s s - l i s t 3   d e n y      a n y
```

```
l i n e   v t y   0   4
```

```
a c c e s s - c l a s s 3   i n
```

```
e x e c - t i m e o u t 5   0
```

```
t r a n s p o r t i n p u t t e l n e t s s h
```

```
t r a n s p o r t o u t p u t n o n e
```

```
t r a n s p o r t p r e f e r r e d n o n e
```

```
p a s s w o r d 7   0 4 5 8 0 2 1 5 0 C 2 E
```


VTY Security

- Use more robust ACLs with the logging feature to spot the probes on you network

```
a c c e s s - l i s t 1 9 9 p e r m i t t c p 1 . 2 . 3 . 0 0 . 0 . 0 . 2 5 5 a n y
```

```
a c c e s s - l i s t 1 9 9 p e r m i t t c p 1 . 2 . 4 . 0 0 . 0 . 0 . 2 5 5 a n y
```

```
a c c e s s - l i s t 1 9 9 d e n y t c p a n y a n y r a n g e 0 6 5 5 3 5 l o g
```

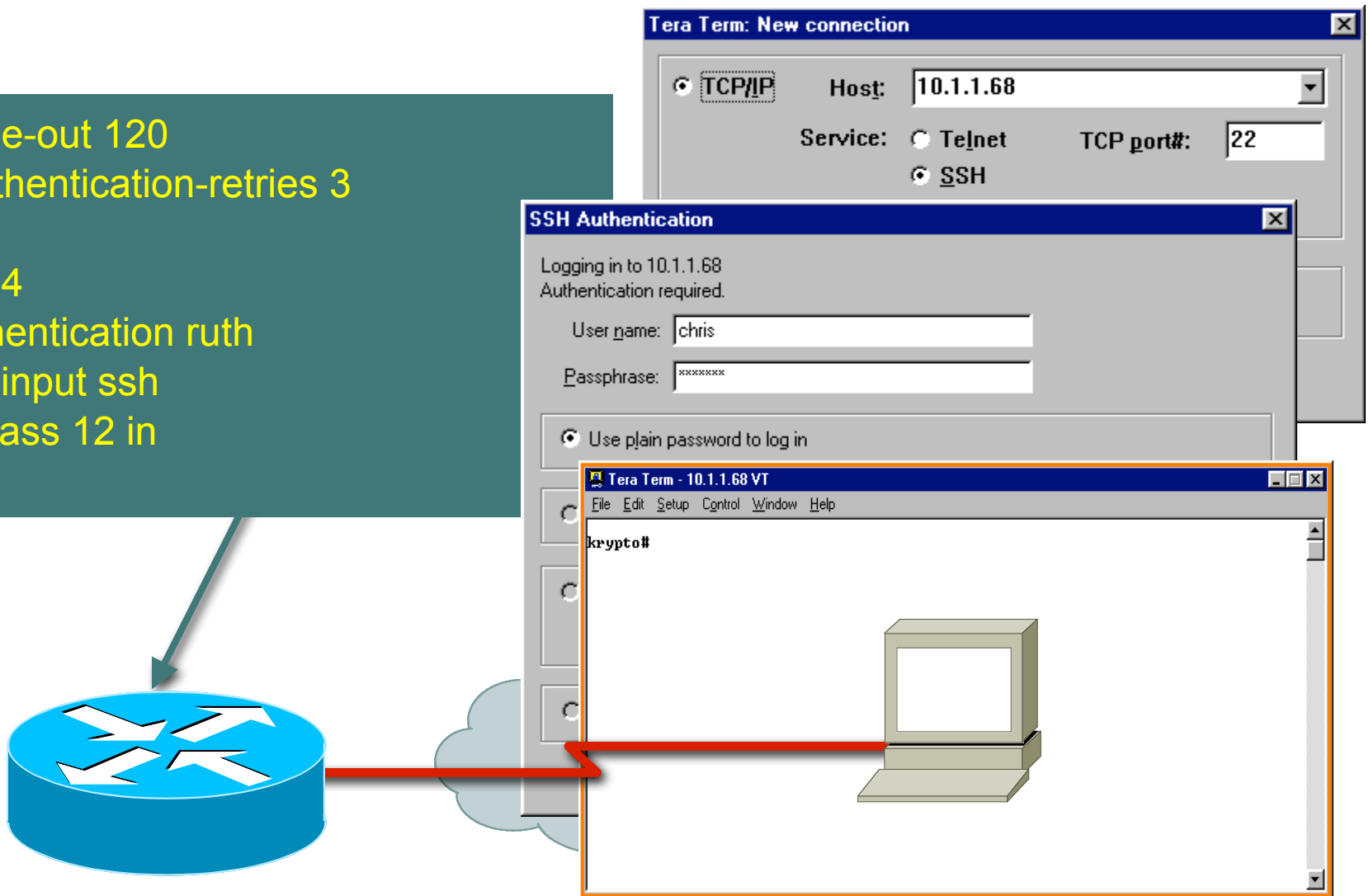
```
a c c e s s - l i s t 1 9 9 d e n y i p a n y a n y l o g
```

SSH Support in ISP Code

- **Cisco IOS Software supports SSHv1 and SSHv2**
- **SSH Server**
Server for remote connections
- **SSH Client**
SSH connections can be generated from router to another router or to the SSH server
- **SCP**
Enables the copying of config and image files to and from the router via SSH
- **SSH Terminal-Line Access (Reverse-SSH)**
Accessing terminal lines of the router via SSH

Cisco IOS SSH Configuration

```
ip ssh time-out 120
ip ssh authentication-retries 3
!
line vty 0 4
login authentication ruth
transport input ssh
access-class 12 in
```



SSH Server Configuration Prerequisites

- The Router's Hostname must be configured:

```
h o s t n a m e < r o u t e r n a m e >
```

- The DNS Resolver on the Router must be configured:

```
i p d o m a i n - n a m e r t p . c i s c o . c o m
```

- Crypto Keys must be configured:

```
c r y p t o k e y g e n e r a t e r s a
```

- Note: When crypto keys are deleted, SSH server is disabled:

```
c r y p t o k e y z e r o i s e r s a
```

SSH Server Configuration

- When crypto key is generated, SSH server is started
- Set up the source interface for outbound SSH connections

```
ip ssh source-interface
```

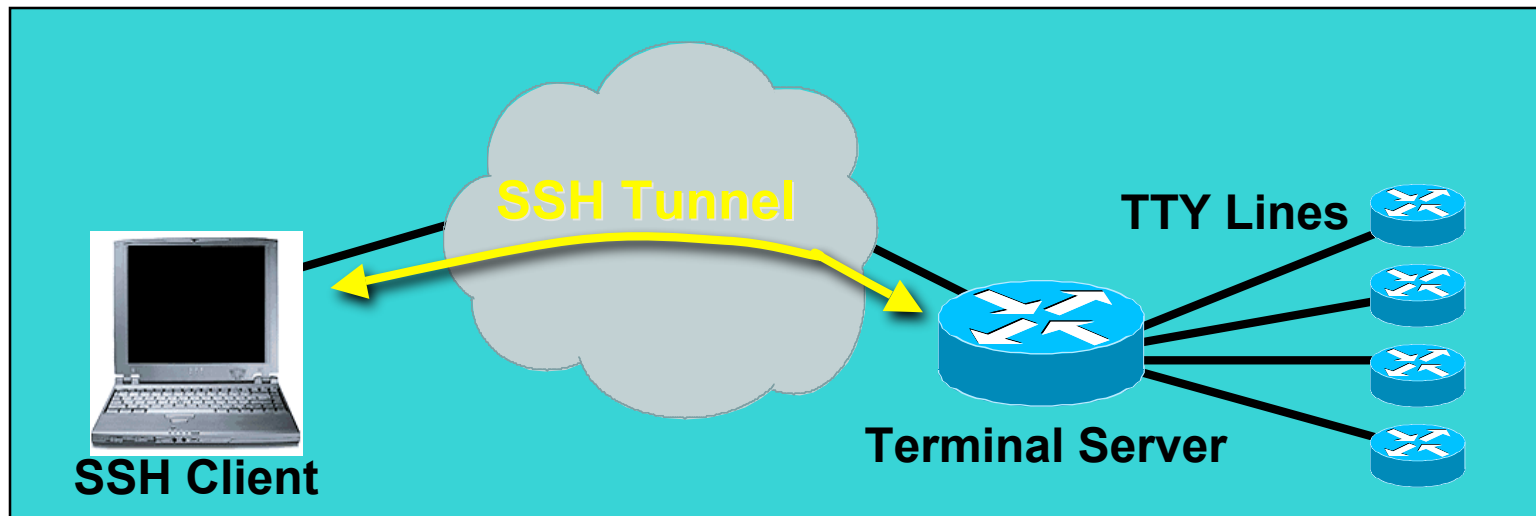
- Add it as input transport

```
line vty 0 4
```

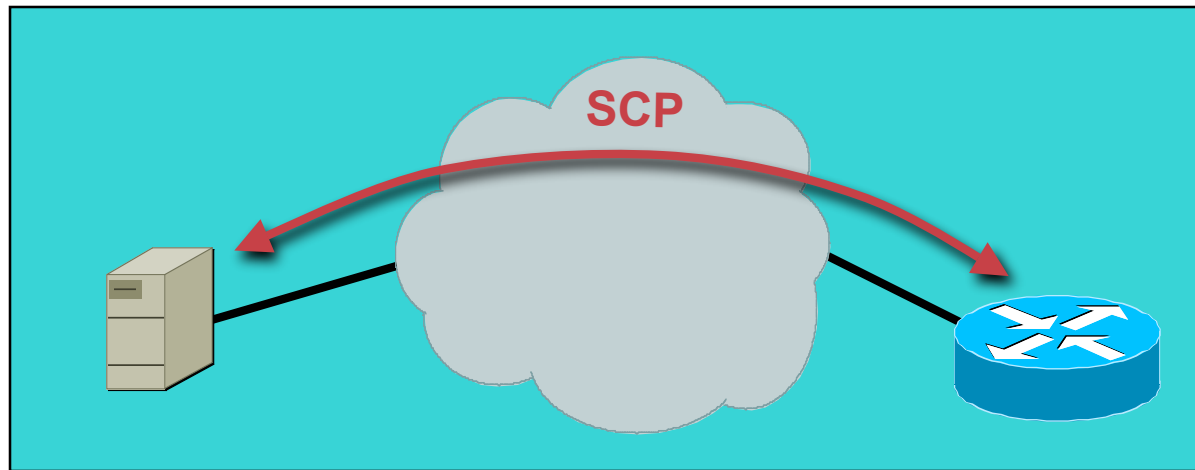
```
transport input telnet ssh
```

SSH Terminal-Line Access

- How can security be increased on Telnet sessions, when using terminal servers for access to console ports in data center
- Included a feature to allow SSH access to lines in async modules, allowing encrypted access to TTY connections



Secure Copy (SCP)



- **Secure File Copy with SSH encryption:**

Router Configs

Router IOS Images

- http://www.cisco.com/en/US/products/sw/iosswrel/ps1839/products_feature_guide09186a0080087b18.html

Secure Copy (SCP)

! Server configuration, client side is always available:

```
aaa authentication login default local
```

```
aaa authorization exec default local
```

```
username pgustafs privilege 15 password 0 lab
```

ip scp server enable

! scp relies on AAA authentication and authorization to function You'll need to configure AAA and enable for SCP server

Cisco IOS CLI dialogue:

```
scp -c 3des c4500-ik2s-mz.scp bstiff@cisco-lab:null:
```

```
bstiff@cisco-lab's password:
```

```
c4500-ik2s-mz.scp | 1072 KB | 67.0 kB/s | ETA: 00:01:24 | 15%
```

Uses SSH encryption hence RSA keys needs to be generated.

(crypto key generate rsa)

What is ISP AAA and ISP AA?

- **ISPs get network infrastructure access and customer access confused!**

Infrastructure access needs to be Authentication, Authorization, and Accounting (AAA) are used secure your network. Accounting is really an Audit function.

(Focus: Operational Accountability.)

Customer access needs to be Authentication and Accounting (AA). Authorization is rarely used.

(Focus: Billing)

- **Endemic problem of ISPs thinking that their existing customer accounting solution will help them secure their network.**

Separate Security Domains!

- **Network Infrastructure Access and Customer Access are separate security domains.**

Network Infrastructure access is controlled by the network operations team.

Customer access is controlled by the customer provisioning team.

- **Requirement to have two AAA solutions on customer edge equipment.**

Radius for customers

TACACS+/Kerberos for Network Infrastructure access.

RADIUS vs. TACACS+ vs Kerberos

	RADIUS	TACACS+	KERBEROS
Uses UDP	X		
Uses TCP		X	X
Encryption	Password Only	All But Header	All But Header
Multiprotocol Support		X	
Router Mgt Acct Control		X	X
Router Mgt Auth Control		X	X
LEAP Support	X		
XAUTH Support	X	X	X

<http://www.cisco.com/warp/public/480/10.html>

What to Configure?

- 1. Local Authentication to provide a failsafe.**
- 2. Authentication with AAA Server with (TACACS+ used in this example).**
- 3. Accounting/Audit with AAA Server**
- 4. Authorization with AAA Server**
- 5. One Time Password Solution**

Simple Staff Authentication and Failsafe

- Username/Passwords on the Router are used as a back-up when the AAA system goes down.

```
aaa new-model

aaa authentication login neteng tacacs+ local

username joe password 7 1 1 0 4 1 8 1 0 5 1 B 1

username jim password 7 0 3 1 7 B 2 1 8 9 5 F E

line vty 0 4

login neteng

access-class 199 in
```

- Remember - Username/password is more resistant to attack than a plain **password**

Staff Accountability & Audit

TACACS+ Provides a Detailed Audit Trail of what Is Happening on the Network Devices

User-Name	Group-cmd		priv-lvl	service	NAS-Portname	task_id	NAS-IP-reason
bgreene	NOC	enable <cr>	0	shell	tty0	4	210.210.51.224
bgreene	NOC	exit <cr>	0	shell	tty0	5	210.210.51.224
bgreene	NOC	no aaa accounting exec Workshop <cr>	0	shell	tty0	6	210.210.51.224
bgreene	NOC	exit <cr>	0	shell	tty0	8	210.210.51.224
pfs	NOC	enable <cr>	0	shell	tty0	11	210.210.51.224
pfs	NOC	exit <cr>	0	shell	tty0	12	210.210.51.224
bgreene	NOC	enable <cr>	0	shell	tty0	14	210.210.51.224
bgreene	NOC	show accounting <cr>	15	shell	tty0	16	210.210.51.224
bgreene	NOC	write terminal <cr>	15	shell	tty0	17	210.210.51.224
bgreene	NOC	configure <cr>	15	shell	tty0	18	210.210.51.224
bgreene	NOC	exit <cr>	0	shell	tty0	20	210.210.51.224
bgreene	NOC	write terminal <cr>	15	shell	tty0	21	210.210.51.224
bgreene	NOC	configure <cr>	15	shell	tty0	22	210.210.51.224
bgreene	NOC	aaa new-model <cr>	15	shell	tty0	23	210.210.51.224
bgreene	NOC	aaa authorization commands 0 default tacacs+ none <cr>	15	shell	tty0	24	210.210.51.224
bgreene	NOC	exit <cr>	0	shell	tty0	25	210.210.51.224
bgreene	NOC	ping <cr>	15	shell	tty0	32	210.210.51.224
bgreene	NOC	show running-config <cr>	15	shell	tty66	35	210.210.51.224
bgreene	NOC	router ospf 210 <cr>	15	shell	tty66	45	210.210.51.224
bgreene	NOC	debug ip ospf events <cr>	15	shell	tty66	46	210.210.51.224

Set Privileges

- **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

Checkpoint with default Authorization

- So now you can have the following:

```
aaa new-model

aaa authentication login default tacacs+ local enable

aaa authentication enable default tacacs+ local enable

aaa authorization exec default tacacs+ local

aaa authorization commands 1 default tacacs+ local

aaa authorization commands 15 default tacacs+ local

aaa accounting exec start-stop tacacs+

ip tacacs source-interface Loopback0

tacacs-server host 215.17.1.1

tacacs-server key CKr3t#

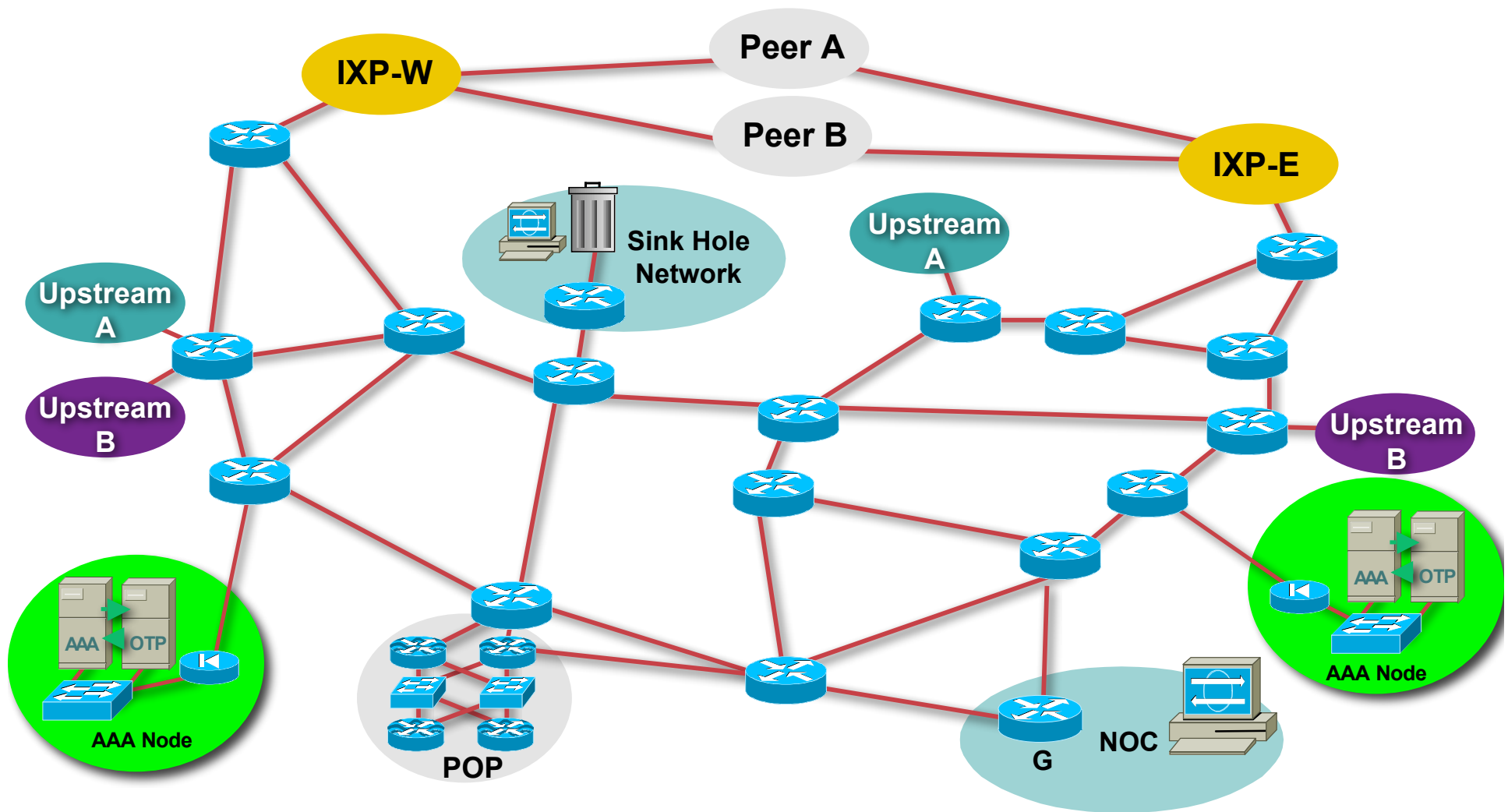
line vty 0 4

 access-class 3 in

username joe privilege 15 password 5 1104181051B1

username jim privilege 1 password 5 0317B21895FE
```


Distribute AAA Servers and Config Backup



TACACS+ Server Redundancy

- **Tries the first. If no response, then moves to the second. If no response then the next, until it fails on TACACS+ and moves to the next AAA option (local in this case).**

```
t a c a c s - s e r v e r h o s t < p r i m a r y   i p >  
t a c a c s - s e r v e r h o s t < s e c o n d a r y   i p >
```

Source Routing

Source Routing

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

`n o i p s o u r c e - r o u t e`

- *traceroute-s* to investigate network failures—valuable tool; but, if you are not using *traceroute-s*, then turn off the feature!

ICMP Unreachable Overload

- Originally, all ICMP Unreachable replies were *punted* from the LC/VIP to the GRP/RP.
- The result was that the GRP/RP's CPU resources could be overloaded, just responding to ICMP Unreachables.
- *Potential Security Hole* that can be used to overload a router.
- Prevented Black Hole Filtering on Router.

ICMP Unreachable Overload

- All Routers who use any static route to Null0 should put *no ip unreachable* (i.e. *BGP Advertisements*).

```
interface Null0
```

```
no ip unreachable
```

```
!
```

```
ip route <dest to drop> <mask> Null0
```

ICMP Unreachable Rate-Limiting

- **New ICMP Unreachable Rate-Limiting Command:**

```
ip icmp rate-limit unreachable [ D F ] < 1 - 4 2 9 4 9 6 7 2 9 5 m i l l i s e c o n d >
```

```
no ip icmp rate-limit unreachable [ d f ]
```

- **Turned on by default and hidden since 12.0(8)S. Default value set to 500 milliseconds.**
- **Peer Review with several top ISP operations engineers are recommending this be set at 2 seconds for normal and DF.**

Tip: scheduler allocate

- **Schedules CPU time spent on processes versus interrupts**

Syntax:

```
s c h e d u l e r a l l o c a t e < i n t e r r u p t > < p r o c e s s e s >
```

< i n t e r r u p t > : **3000-60000 Microseconds handling network interrupts**

< p r o c e s s e s > : **1000-8000 Microseconds running processes**

Example:

```
r o u t e r ( c o n f i g ) # s c h e d u l e r a l l o c a t e 8 0 0 0 8 0 0 0
```

**Very useful under heavy load!
Recommended Standard Config!**

Input Hold Queue

Input Hold Queue

- **This is the queue that stores packets destined for the router.**
- **If there are too many packets, the route stores them in the input hold queue.**
- **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)**
- **Applied to all interfaces**

```
i n t e r f a c e   x x x x x x
```

```
h o l d - q u e u e   1 5 0 0   i n
```

Input Hold Queue

```
1 2 0 0 8 -e 1 0 -2 # sh inter pos 5 / 0

P O S 5 / 0 is up , line protocol is up

.

O u t p u t q u e u e 0 / 4 0 , 0 d r o p s ; i n p u t q u e u e 9 7 / 1 5 0 0 , 5 4 d r o p s

5 m i n u t e i n p u t r a t e 7 6 5 0 2 0 0 0 b i t s / s e c , 3 1 1 3 9 p a c k e t s / s e c

5 m i n u t e o u t p u t r a t e 7 2 5 1 7 0 0 0 b i t s / s e c , 2 6 5 6 0 p a c k e t s / s e c

.

.
```

26Mbps DOS on port 179 – non-successful spoof

Selective Packet Discard

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

```
ip spd enable ( 1 1 . 1 C A & C C )
```

Selective Packet Discard (SPD)

- **Software Switching – SPD allows Control & Management Plane Traffic destined for the router to not get dropped when a circuit gets saturated.**
- **ASIC Switching – SPD allows a deeper buffer for the Control & Management Plane Traffic destined for the router – added resistance to direct DOS Attacks and buffer room for surges in control plane traffic (i.e. times of convergence).**

Selective Packet Discard

- **Recommended Settings:**

`ip s p d h e a d r o o m 1 0 0 0` **Default is 100.**

Experience shows that the higher settings help.

`ip s p d m o d e a g g r e s s i v e` **Does not work on the GSR
– but does on other platforms.**

Using Receive ACL (rACL)

Receive ACLs - Overview

- Excessive traffic destined to GRP can lead to high CPU → DoS
- Receive ACLs filter traffic destined to the GRP via receive adjacencies
- rACLs explicitly permit or deny traffic destined to the GRP
- rACL do NOT affect transit traffic
- Traffic is filtering on the ingress LC, prior to GRP 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**

rACL – Required Entries

- **Routing Protocols**
OSPF, EIGRP, etc.
IS-IS is not filtered by rACL
- **Remote Access**
telnet, ssh
- **Management**
SNMP

rACL – Required Entries

- **TACACS+/Radius**

AAA protocols

- **NTP**

Time sync

- **Others that you use?**

Do you need traceroute from the router?

Other applications not covered here?

Administrative and Operational Practices

Loopback Interface

- **Most ISPs make use of the router loopback interface**
- **IP address configured is a host address**
- **Configuration example:**

```
interface loopback 0

description Loopback Interface of CORE-GW 3

ip address 215.18.3.34 255.255.255.255

no ip redirects
```

Configuration Management

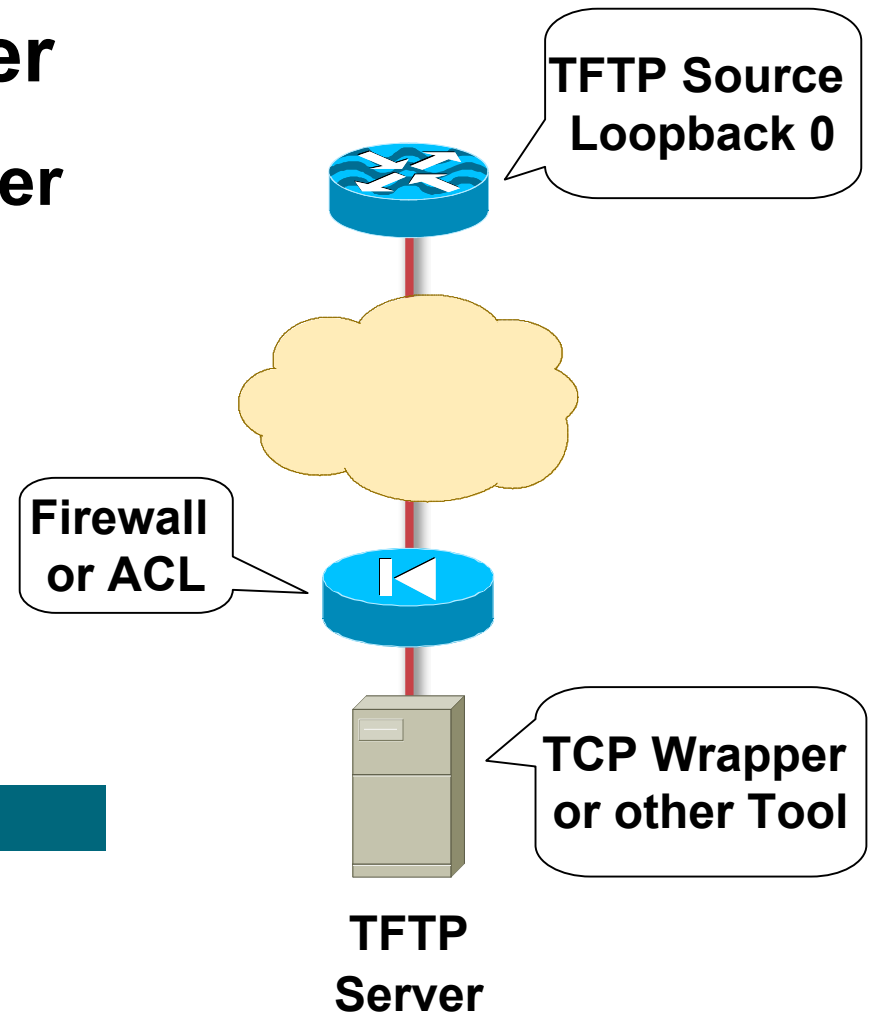
- **Backup NVRAM configuration off the router:**
 - Write configuration to TFTP server**
 - TFTP server files kept under revision control**
 - Router configuration built from master database**
- **Allows rapid recovery in case of emergency**

Configuration Management

- **Secure the TFTP server**
TFTP loopback 0 on router
Firewall/ACL

**Wrapper on TFTP server
which only allows the
router's loopback
address**

```
ip tftp source-interface Loopback 0
```



FTP Client Support

- **TFTP has its limitations**
- **FTP client support is added in IOS 12.0; this allows for FTP upload/downloads**
- **Remember to use the same security/redundancy options with loopback 0:**

```
ip ftp source-interface loopback 0
```

Use Detailed Logging

- Off load logging information to a logging server
- Use the full detailed logging features to keep exact details of the activities

```
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone

logging buffered 16384

logging trap debugging

logging facility local7

logging 169.223.32.1

logging 169.223.55.37

logging source-interface loopback0

no logging console | Recommended - keeps the console port free
```

Network Time Protocol

- If you want to cross compare logs, you need to synchronize the time on all the devices
- Use NTP

From external time source

Upstream ISP, Internet, GPS, atomic clock

From internal time source

Router can act as *stratum 1* time source

Network Time Protocol

- **Set timezone**

```
clock timezone <name> [+ / - hours [mins]]
```

- **Router as source**

```
ntp master 1
```

- **External time source (master)**

```
ntp server a.b.c.d
```

- **External time source (equivalent)**

```
ntp peer e.f.g.h
```

Network Time Protocol

- **Example configuration:**

```
c l o c k   t i m e   z o n e   A E S T   1   0
```

```
n t p   u p d a t e   - c a l e n d a r
```

```
n t p   s o u r c e   l o o p b a c k 0
```

```
n t p   s e r v e r   < o t h e r   t i m e   s o u r c e   >
```

```
n t p   p e e r   < o t h e r   t i m e   s o u r c e   >
```

```
n t p   p e e r   < o t h e r   t i m e   s o u r c e   >
```

Network Time Protocol

- **Network Time Protocol (NTP) used to synchronize the time on all the devices**
- **NTP packets leave router with loopback address as source**
- **Configuration example:**

```
n t p   s o u r c e   l o o p b a c k 0
```

```
n t p   s e r v e r   1 6 9 . 2 2 3 . 1 . 1   s o u r c e   l o o p b a c k 1
```

SNMPv1

- Remove any SNMP commands if SNMP is not going to be used
- If SNMP is going to be used:

```
a c c e s s - l i s t 9 8 p e r m i t 1 6 9 . 2 2 3 . 1 . 1
```

```
a c c e s s - l i s t 9 8 d e n y a n y
```

```
s n m p - s e r v e r c o m m u n i t y 5 n m c 0 2 m R O 9 8
```

```
s n m p - s e r v e r t r a p - s o u r c e L o o p b a c k 0
```

```
s n m p - s e r v e r t r a p - a u t h e n t i c a t i o n
```

```
s n m p - s e r v e r h o s t 1 6 9 . 2 2 3 . 1 . 1 5 n m c 0 2 m
```


HTTP Server

- HTTP server in Cisco IOS from 11.1CC and 12.0S

Router configuration via web interface

- **Disable** if not going to be used (disabled by default):

```
no ip http server
```

- **Configure securely** if going to be used:

```
ip http server
```

```
ip http port 8765
```

```
ip http authentication aaa
```

```
ip http access-class < 1 - 9 9 >
```

Core Dumps

- **Cisco routers have a core dump feature that will allow ISPs to transfer a copy of the core dump to a specific FTP server**
- **Set up a FTP account on the server the router will send the core dump to**
- **The server should NOT be a public server**
 - Use filters and secure accounts**
 - Locate in NOC with NOC staff access only**
 - Enough disk space to handle the dumps**

Core Dumps

- **Example configuration:**

```
i p  f t p  u s e r n a m e  c i s c o
```

```
i p  f t p  p a s s w o r d  7  0  4  5  8  0  2  1  5  0  C  2  E
```

```
i p  f t p  s o u r c e - i n t e r f a c e  l o o p b a c k  0
```

```
e x c e p t i o n  p r o t o c o l  f t p
```

```
e x c e p t i o n  d u m p  1  6  9  . 2  2  3  . 3  2  . 1
```

Netflow

- **Provides network administrators with “packet flow” information**
- **Allows:**
 - Security monitoring**
 - Network management and planning**
 - Customer billing**
 - Traffic flow analysis**
- **Available from 11.1CC for 7x00 and 12.0 for remaining router platforms**

Netflow

- **Configuration example:**

```
interface serial 5 / 0
```

```
ip route-cache flow | Prior to IOS 12.4
```

```
ip flow [ingress | egress] | From IOS 12.4
```

- **If CEF not configured, Netflow enhances existing switching path (i.e. optimum switching)**
- **If CEF configured, Netflow becomes a flow information gatherer and feature acceleration tool**

Netflow

- **Information export:**

Router to collector system

```
ip flow-export version 5 [origin-as peer-as]
```

```
ip flow-export destination x.x.x.x <udp-port>
```

- **Flow aggregation (new in 12.0S):**

Router sends aggregate records to collector system

```
ip flow-aggregation cache aslpre fixldestlsourcelproto
```

```
enabled
```

```
export destination x.x.x.x <udp-port>
```

Netflow—Simple Monitoring

- Sample output on router:

```

Beta-7200-2>sh ip cache flow
IP packet size distribution (14280M total packets):

 1 -32    64    96   128   160   192   224   256   288   320   352   384   416   448   480
.000 .145 .403 .101 .178 .105 .017 .005 .003 .001 .000 .000 .000 .000 .001

 512   544   576 1024 1536 2048 2560 3072 3584 4096 4608
.001 .000 .025 .001 .004 .000 .000 .000 .000 .000 .000

IP Flow Switching Cache, 4456704 bytes
 14369 active, 51167 inactive, 253731473 added
 1582853980 aging polls, 0 flow alloc failures
last clearing of statistics 16w 5d

Protocol      Total    Flows    Packets Bytes    Packets Active(Sec) Idle(Sec)
-----
Flows      /Sec      /Flow  /Pkt      /Sec      /Flow      /Flow
TCP-Telnet    28284      0.0      36    71      0.2      13.4      17.7
TCP-FTP       171390     0.0      15     63     0.6       8.1      16.6
TCP-FTPD      104030     0.0     693    384     16.8     29.7       9.7
TCP-WWW       28119533    6.5      17   290     115.8     6.5     10.9
TCP-SMTP      3615725     0.8      18   266     15.7      5.6     15.5
TCP-X         1649       0.0       3     84      0.0       4.1     14.0
TCP-BGP       1483900     0.3       5   258      1.7     13.1     19.1
TCP-NNTP      2330       0.0       2    53      0.0      8.4     20.7
TCP-Frag       484       0.0       1    46      0.0      1.2     20.9
TCP-other     343437823   79.9       5  129    410.9     2.5     11.0

```

Netflow—Simple Monitoring

- Sample output on router (continued):

```

Protocol      Total   Flow s   Packets Bytes   Packets Active (Sec) Idle (Sec)
-----
Flow s      / Sec      / Flow / Pkt      / Sec      / Flow      / Flow

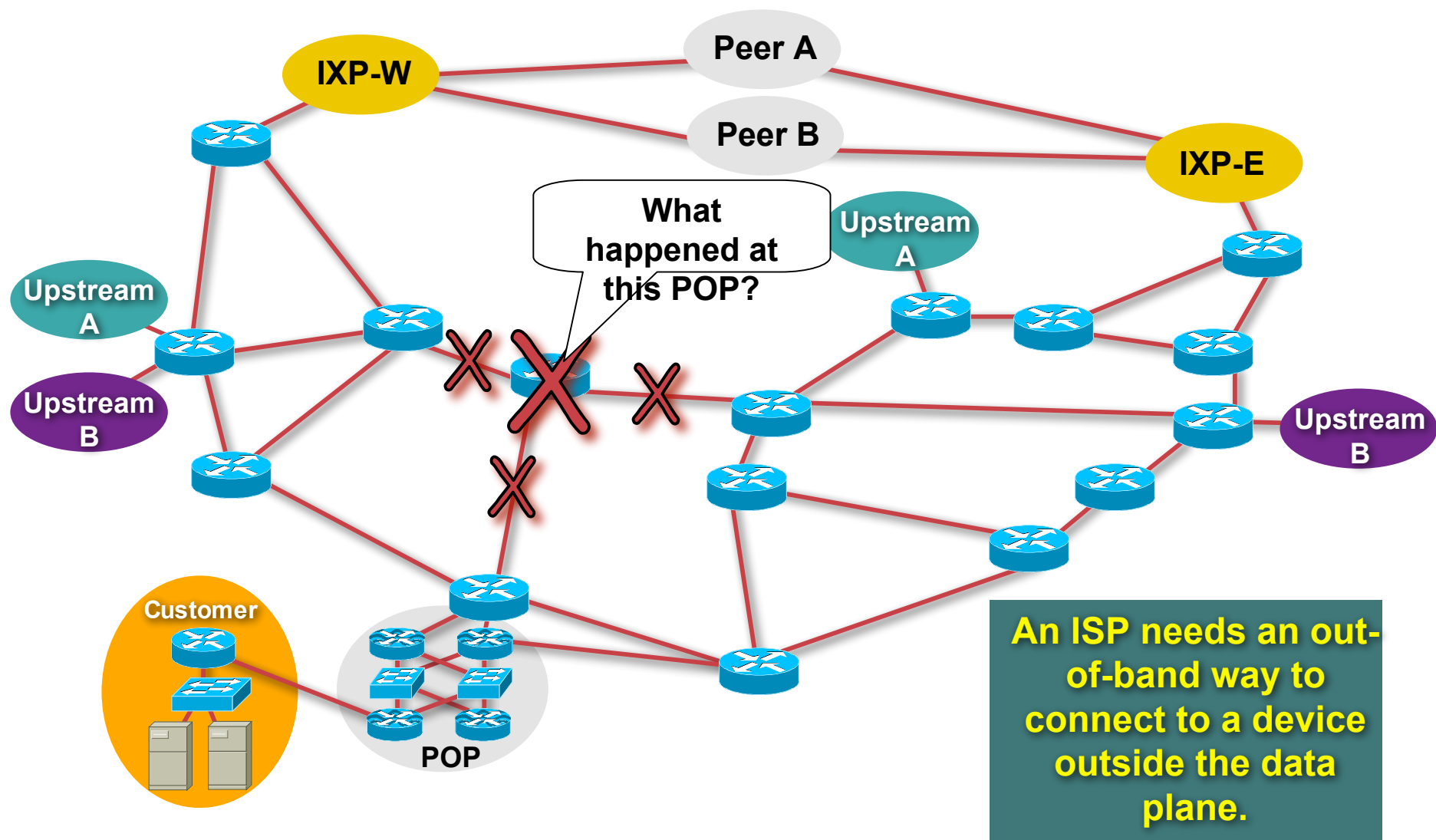
UDP-DNS       2 5 1 3 1 4 0 6 9 4      5 8 5 .1      3      9 0      1 7 7 8 .6      5 .3      2 1 .5
UDP-NTTP      2 6 7 5 2 0 3      0 .6      1      7 6      0 .6      0 .0      2 1 .6
UDP-TFTP      2 5 7 5 0      0 .0      6      1 5 7      0 .0      2 0 .1      2 0 .8
UDP-Frag      7 3 7      0 .0      5      2 1 0      0 .0      1 4 .4      2 1 .4
UDP-other    1 5 3 2 6 7 7 3 0 2      3 5 6 .8      2      1 5 4      9 5 0 .7      4 .3      2 1 .6
ICMP          3 0 7 8 4 3 9 2      7 .1      4      1 0 9      3 0 .7      7 .3      2 0 .5
IGMP          3 1      0 .0      1 9 0 3      1 0 8 5      0 .0      8 9 .7      2 1 .7
IP-other     9 8 5 0 8 1      0 .2      8      3 5 4      1 .9      1 3 .9      2 0 .2
Total:      4 4 5 7 2 5 4 3 3 8      1 0 3 7 .7      3      1 2 3      3 3 2 4 .8      4 .8      2 0 .6

Src If      Src IP address      Dst If      Dst IP address      Pr Src P Dst P Pkts
Se 2 / 0      2 0 3 .1 6 1 .2 3 4 .2 1 1 Fa 1 / 0      2 0 3 .3 7 .2 5 5 .9 7      1 1 0 4 0 4 0 0 3 5      1
Fa 1 / 0      2 0 3 .3 7 .2 5 5 .9 7      Se 2 / 0      2 0 3 .1 6 1 .2 3 4 .2 1 1 1 1 0 0 3 5 0 4 0 4      1
Fa 1 / 0      2 0 3 .3 7 .2 5 5 .9 7      Se 2 / 0      2 0 3 .9 3 .1 1 1 .1      1 1 0 0 3 5 8 1 2 4      1
Fa 1 / 0      2 0 3 .3 7 .2 5 5 .1 1 4      Se 2 / 0      1 9 5 .6 7 .2 0 8 .2 4 8      1 1 1 B 3 A 3 F 0 4      4 6 7 5
Se 2 / 0      1 9 5 .6 7 .2 0 8 .2 4 8      Fa 1 / 0      2 0 3 .3 7 .2 5 5 .1 1 4      1 1 3 F 0 4 1 B 3 A      6 6 7 2
Se 2 / 0      2 0 3 .9 3 .1 1 1 .1      Fa 1 / 0      2 0 3 .3 7 .2 5 5 .9 7      1 1 8 1 2 4 0 0 3 5      1
Fa 1 / 0      2 0 3 .3 7 .2 5 5 .9 7      Se 2 / 0      2 0 3 .1 3 2 .2 2 4 .1 1      1 1 0 0 3 5 0 E D C      1
Se 2 / 0      2 1 6 .1 5 4 .2 4 0 .8      Fa 1 / 0      2 0 3 .3 7 .2 5 5 .9 7      1 1 0 4 2 4 0 0 3 5      1 2 K
Fa 1 / 0      2 0 3 .3 7 .2 5 5 .9 7      Se 2 / 0      2 1 6 .1 5 4 .2 4 0 .8      1 1 0 0 3 5 0 4 2 4      1 2 K
Se 2 / 0      2 0 3 .1 3 2 .2 2 4 .1 1      Fa 1 / 0      2 0 3 .3 7 .2 5 5 .9 7      1 1 0 E D C 0 0 3 5      1
...etc...

```


Out of Band Management

Router Crash? Cable Cut? DOS?



Traditional Reverse Telnet OOB

- **OOB example—Access Server with reverse telnet:**

Modem attached to the access server to allow NOC dial in in case of total POP isolation

Console ports of all network equipment connected to async ports of the access server – NOC reverse telnets through the async ports into the console of the POP.

Access server's LAN and/or WAN link connects to network core (least preferred) or via separate management network to NOC

- ***Full remote control access under all circumstances***

Phase 1 – Preparation for the Attack

Securing the Routing Protocol and Control Plane

Securing the Routing Protocol

Routing Protocol Security

- **Routing protocol can be attacked**

Denial of service

Smoke screens

False information

Reroute packets

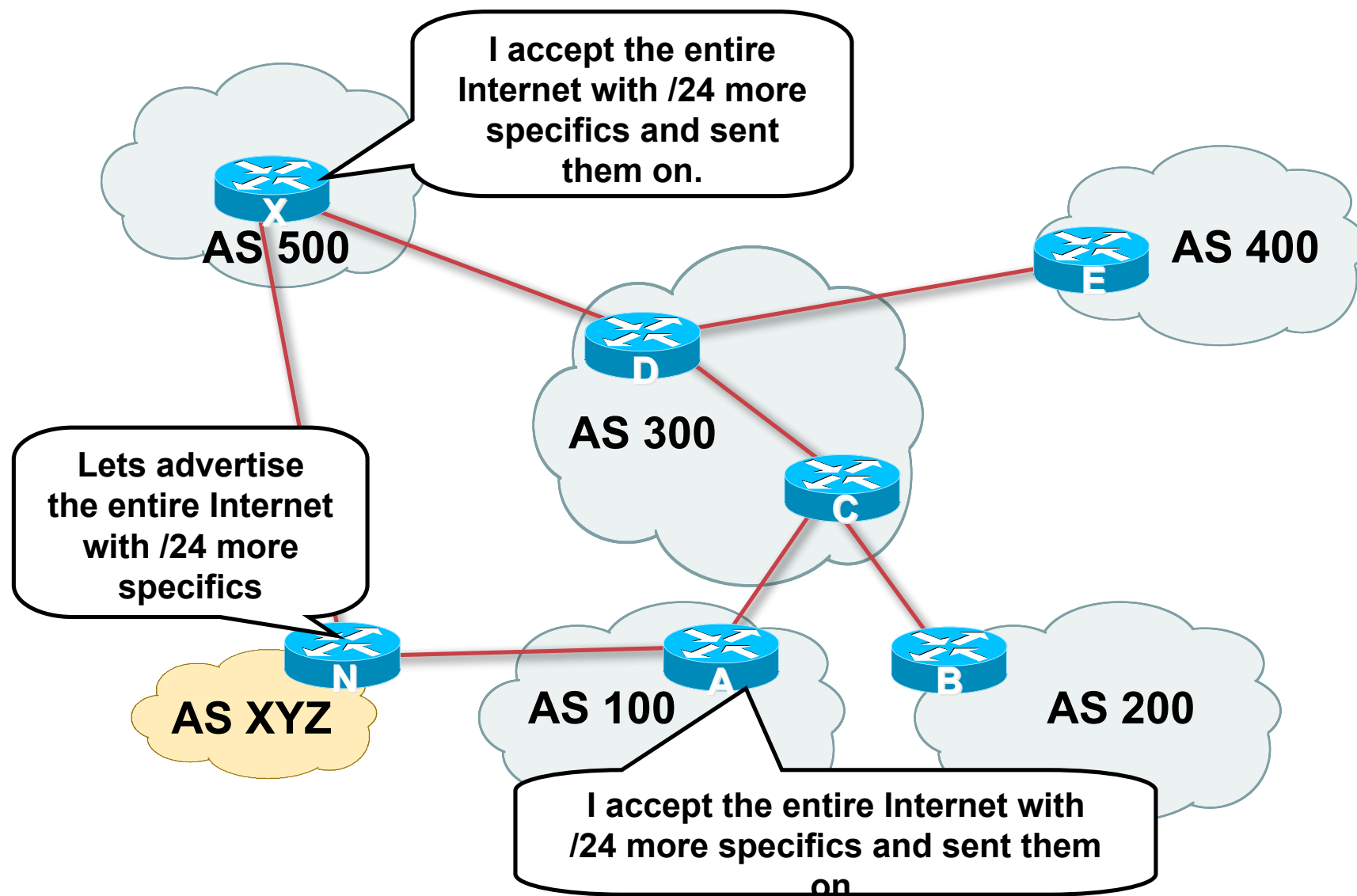
May Be Accidental or Intentional

Agenda

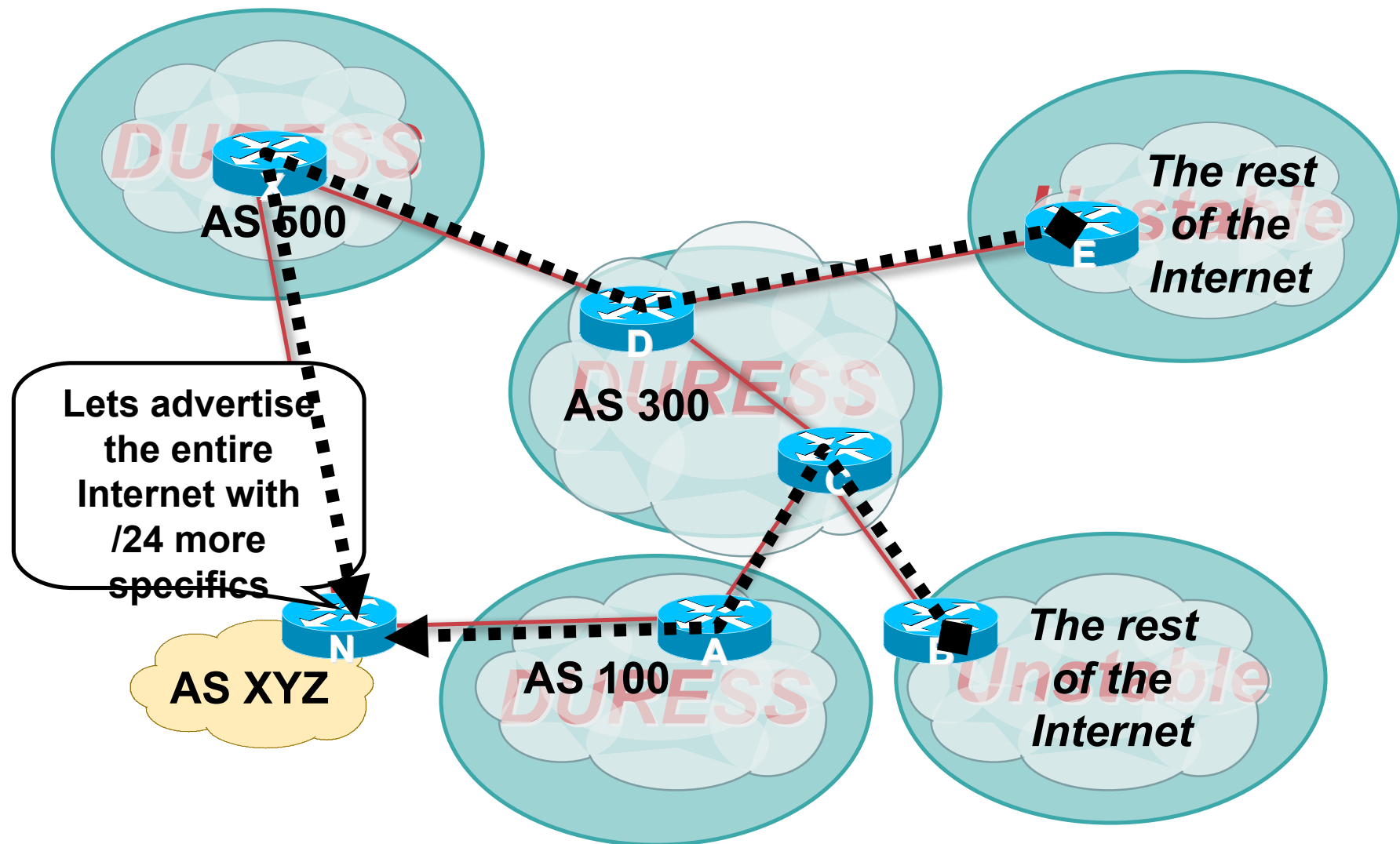
- **Why to Prefix Filter? (Threats)**
- **How to Prefix Filter?**
- **Where to Prefix Filter?**
- **Prefix Filter on Customers**
- **Egress Filter to Peers**
- **Ingress Filter from Peers**
- **Protocol Authentication (MD5)**
- **BGP BCPs that help add Resistance**

Garbage In – Garbage Out Threat

Garbage in – Garbage Out: What is it?

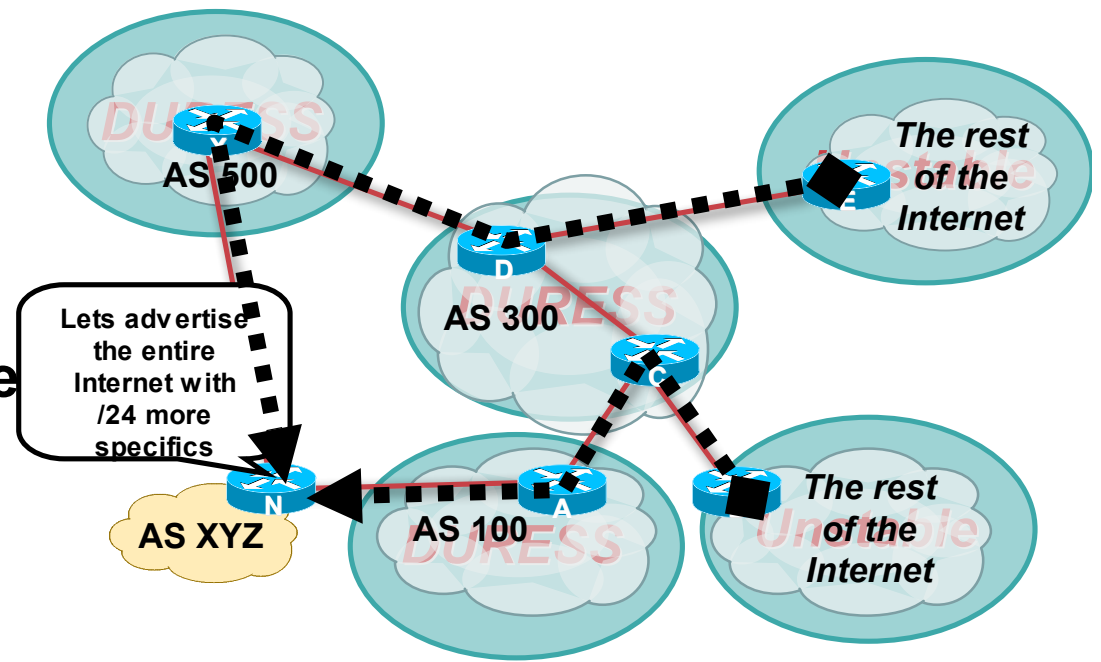


Garbage in – Garbage Out: Results



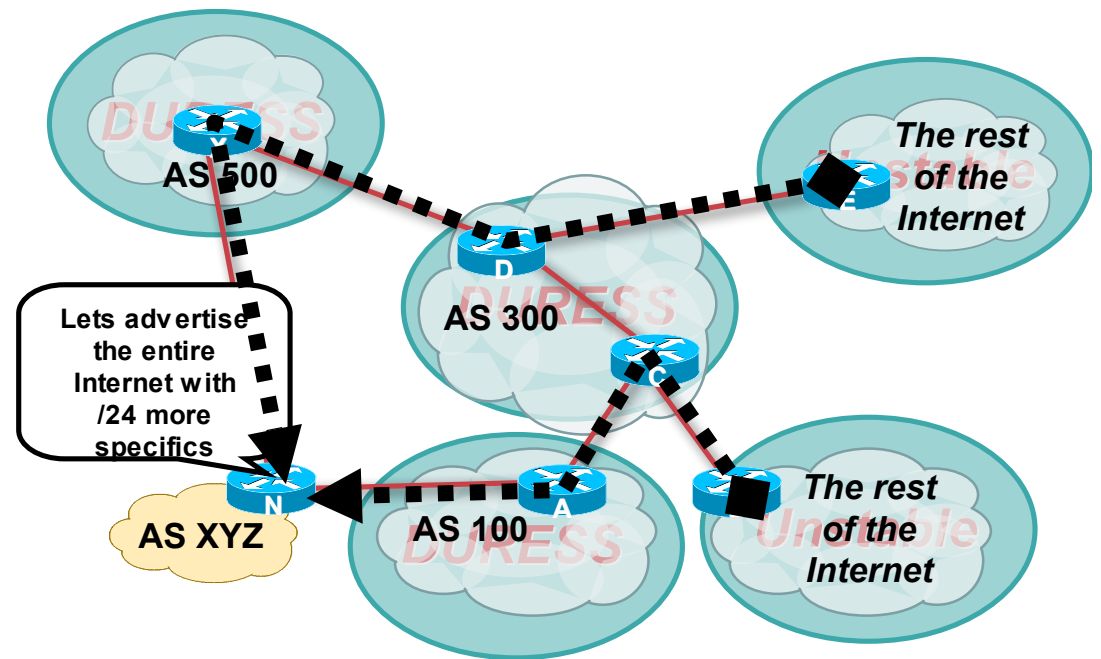
Garbage in – Garbage Out: Impact

- Garbage in – Garbage out does happen on the Net
- AS 7007 Incident (1997) was the most visible case of this problem.
- Key damage are to those ISPs who pass on the garbage.
- Disruption, Duress, and Instability has been an Internet wide effect of Garbage in – Garbage out.



Garbage in – Garbage Out: What to do?

- Take care of your own Network.
 - Filter your customers
 - Filter you advertisements
- Net Police Filtering
 - Mitigate the impact when it happens
- Prefix Filtering and Max Prefix Limits



How to Prefix Filter?

Ingress and Egress Route Filtering

- **Two **flavours** of prefix filtering:**
 - Distribute list—Now obsolete**
 - Prefix list—Widely used, higher performance**
- **Two filtering techniques:**
 - Explicit Permit (permit then deny any)**
 - Explicit Deny (deny then permit any)**

Ingress and Egress Route Filtering

Prefix-List for a for a BGP Prefix List

```
ip prefix-list deny-sua deny 0.0.0.0 / 8 le 32

ip prefix-list deny-sua deny 10.0.0.0 / 8 le 32

ip prefix-list deny-sua deny 127.0.0.0 / 8 le 32

ip prefix-list deny-sua deny 169.254.0.0 / 16 le 32

ip prefix-list deny-sua deny 172.16.0.0 / 12 le 32

ip prefix-list deny-sua deny 192.0.2.0.0 / 24 le 32

ip prefix-list deny-sua deny 192.168.0.0 / 16 le 32

ip prefix-list deny-sua deny 224.0.0.0 / 3 le 32

ip prefix-list deny-sua permit 0.0.0.0 / 0 le 32
```

Ingress and Egress Route Filtering

BGP with Prefix-List Flavour of Route Filtering

```
router bgp 200

no synchronization

neighbor 220.220.4.1 remote-as 210

neighbor 220.220.4.1 version 4

neighbor 220.220.4.1 prefix-list deny-sua in

neighbor 220.220.4.1 prefix-list deny-sua out

neighbor 222.222.8.1 remote-as 220

neighbor 222.222.8.1 version 4

neighbor 222.222.8.1 prefix-list deny-sua in

neighbor 222.222.8.1 prefix-list deny-sua out

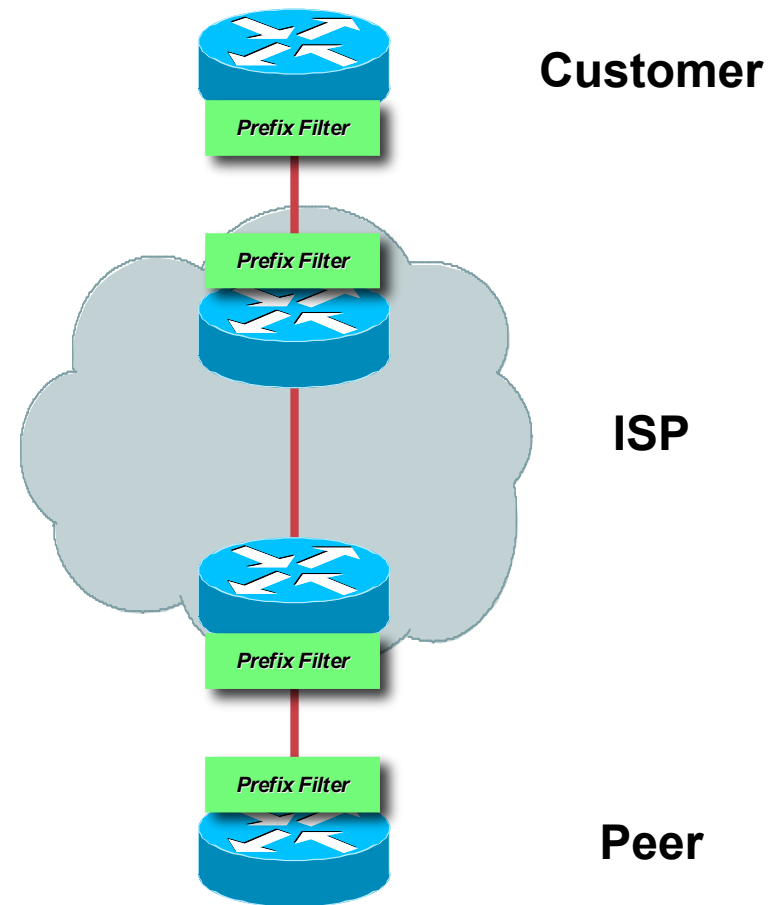
no auto-summary

!
```


Where to Prefix Filter?

Where to Prefix Filter?

- Customer's Ingress/Egress
- ISP Ingress on Customer (may Egress to Customer)
- ISP Egress to Peer and Ingress from Peer
- Peer Ingress from ISP and Egress to ISP



What to Prefix Filter?

**Special Use Addresses
(RFC3330) and Bogons**

Special Use Addresses

- **There are routes that should NOT be routed on the Internet**

RFC 1918 and “Martian” networks

127.0.0.0/8 and multicast blocks

Certain RFC3330 addresses:

<http://www.rfc-editor.org/rfc/rfc3330.txt>

- **BGP should have filters applied so that these routes are not advertised to or propagated through the Internet**

Special Use Addresses

- **Quick review**

0.0.0.0/8 and 0.0.0.0/32—Default and broadcast

127.0.0.0/8—Host loopback

192.0.2.0/24—TEST-NET for documentation

**10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16—RFC
1918 private addresses**

169.254.0.0/16—End node auto-configuration

Special Use Addresses

```
ip prefix-list deny-sua deny 0.0.0.0 / 8 le 32
```

```
ip prefix-list deny-sua deny 10.0.0.0 / 8 le 32
```

```
ip prefix-list deny-sua deny 127.0.0.0 / 8 le 32
```

```
ip prefix-list deny-sua deny 169.254.0.0 / 16 le 32
```

```
ip prefix-list deny-sua deny 172.16.0.0 / 12 le 32
```

```
ip prefix-list deny-sua deny 192.0.2.0 / 24 le 32
```

```
ip prefix-list deny-sua deny 192.168.0.0 / 16 le 32
```

```
ip prefix-list deny-sua deny 224.0.0.0 / 3 le 32
```

```
ip prefix-list deny-sua deny 0.0.0.0 / 0 ge 25
```

```
ip prefix-list deny-sua permit 0.0.0.0 / 0 le 32
```

Bogons

- IANA has published the blocks of IPv4 addresses that have been allocated to the RIRs or directly to end-users:

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

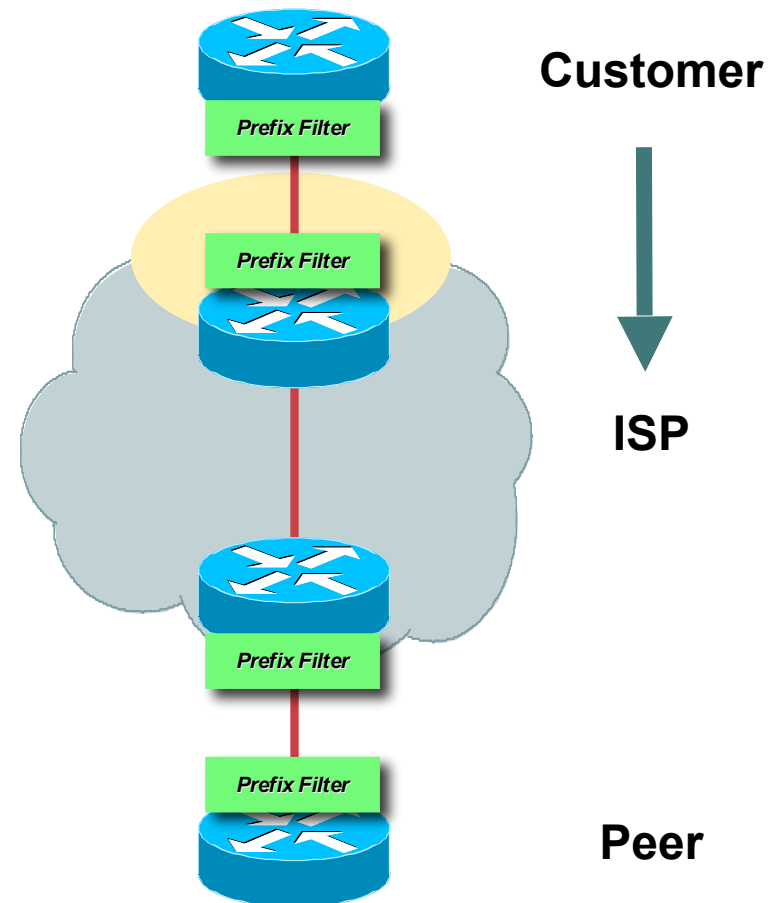
- Only these blocks of IPv4 addresses should be advertised into the global Internet Route Table.
- Filters should be applied on the AS border to block all other address space
- Consider using the Project Cymru bogon BGP feed

<http://www.cymru.com/BGP/bogon-rs.html>

Prefix Filters on Customers

Prefix Filters on Customers

- **Prefix filter all routes from your customers!**



Receiving Customer Prefixes

- **ISPs should only accept prefixes which have been assigned or allocated to their downstream peer/customer.**
- **For example**
 - Downstream has 220.50.0.0/20 block**
 - Should only announce this to peers**
 - Peers should only accept this from them**
 - Explicitly permit prefixes from other ISPs (i.e. multihomed to two or more ISPS).**

Receiving Customer Prefixes

- **Configuration example on upstream:**

```
router bgp 100

neighbor 2.2.2.2 2.2.2.10.1 remote-as 101

neighbor 2.2.2.2 2.2.2.10.1 prefix-list customer in

!

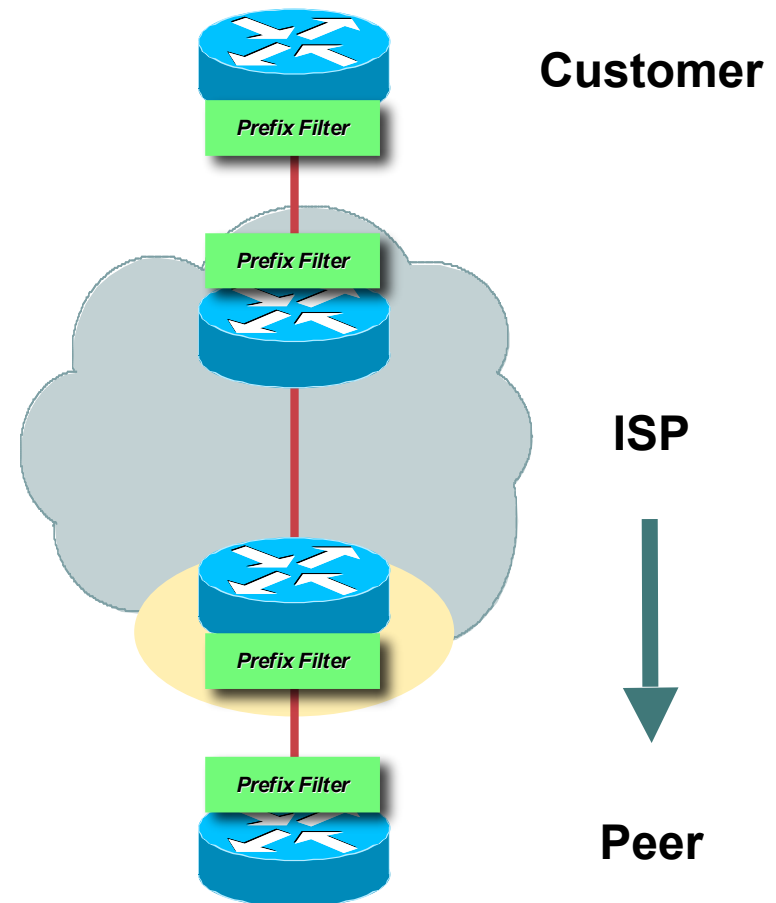
ip prefix-list customer permit 2.2.0.50.0.0 / 2

ip prefix-list customer deny 0.0.0.0 / 0 le 32
```

Prefixes to Peers

Prefixes to Peers

- Prefix filter all routes to your peers!



Prefixes to Peers

- **What do you send to the Internet?**

Your prefixes.

More specific customers prefixes (customers who are multihoming)

- **What do you not send to the Internet?**

RFC3330 Prefixes – assume junk will leak into your iBGP.

Bogons – assume garbage will leak into your iBGP.

Lower Prefix Boundary – Unless absolutely necessary, do not allow anything in the /25 - /32 range.

Egress Filter to ISP Peers - Issues

- **The egress filter list can grow to be very large:**

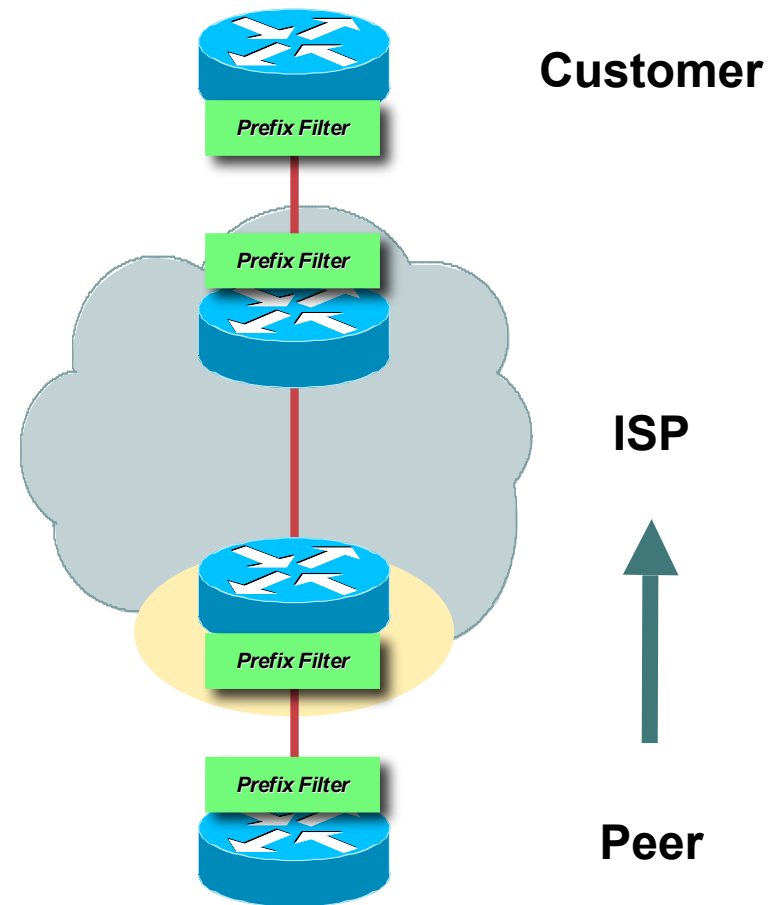
More specifics for customers.

Specific blocks from other ISPs

Ingress Prefix Filtering from Peers

Prefixes from Peers

- Prefix filter all routes from your peers!



Ingress Routes from Peers or Upstream

- **Ingress Routes from Peers and/or the Upstream ISP are the nets of the Internet.**
- **Ideally, the peering policy should be specific so that exact filters can be put in place.**

Dynamic nature of the peering makes it hard to maintain specific route filters.

Receiving Prefixes from Upstream & Peers (ideal case)

Don't accept RFC1918 etc prefixes

Don't accept your own prefix

Don't accept default (unless you need it)

Don't accept prefixes longer than /24

Consider *Net Police* Filtering

Receiving Prefixes — Cisco IOS

```
router bgp 100

 network 221.10.0.0 mask 255.255.224.0

 neighbor 221.5.7.1 remote-as 101

 neighbor 221.5.7.1 prefix-list in-filter in

!

ip prefix-list in-filter deny 0.0.0.0 / 0                ! Block default

ip prefix-list in-filter deny 0.0.0.0 / 8 le 32

ip prefix-list in-filter deny 10.0.0.0 / 8 le 32

ip prefix-list in-filter deny 127.0.0.0 / 8 le 32

ip prefix-list in-filter deny 169.254.0.0 / 16 le 32

ip prefix-list in-filter deny 172.16.0.0 / 12 le 32

ip prefix-list in-filter deny 192.0.2.0 / 24 le 32

ip prefix-list in-filter deny 192.168.0.0 / 16 le 32

ip prefix-list in-filter deny 221.10.0.0 / 19 le 32 ! Block local prefix

ip prefix-list in-filter deny 224.0.0.0 / 3 le 32

ip prefix-list in-filter deny 0.0.0.0 / 0 ge 25          ! Block prefixes > / 24

ip prefix-list in-filter permit 0.0.0.0 / 0 le 32
```

Using AS-PATH filters

Using AS-PATH filters

- **Filter routes based on AS path**
Inbound or Outbound
- **Example Configuration:**

```
router bgp 100

network 105.7.0.0 mask 255.255.0.0

neighbor 102.10.1.1 filter-list 5 out

neighbor 102.10.1.1 filter-list 6 in

!

ip as-path access-list 5 permit ^200$

ip as-path access-list 6 permit ^150$
```

AS-Path Regular Expressions

- **Like Unix regular expressions**
 - .** Match one character
 - *** Match any number of preceding expression
 - +** Match at least one of preceding expression
 - ^** Beginning of line
 - \$** End of line
 - _** Beginning, end, white-space, brace
 - |** Or
 - ()** brackets to contain expression

AS-Path Regular Expressions

- **Simple Examples**

.*	match anything
.+	match at least one character
^\$	match routes local to this AS
_1800\$	originated by AS1800
^1800_	received from AS1800
1800	via AS1800
_790_1800_	via AS1800 and AS790
(1800)+	multiple AS1800 in sequence (used to match AS-PATH prepends)

AS-Path Regular Expressions

- Not so simple Examples

`^[0-9]+$`

Match AS_PATH length of one

`^[0-9]+_[0-9]+$`

Match AS_PATH length of two

`^[0-9]*_[0-9]+$`

Match AS_PATH length of one or two

`^[0-9]*_[0-9]*$`

**Match AS_PATH length of one or two
(will also match zero)**

`^[0-9]+_[0-9]+_[0-9]+$`

Match AS_PATH length of three

`_(701|1800)_`

**Match anything which has gone
through AS701 or AS1800**

`_1849(_.+_)12163$`

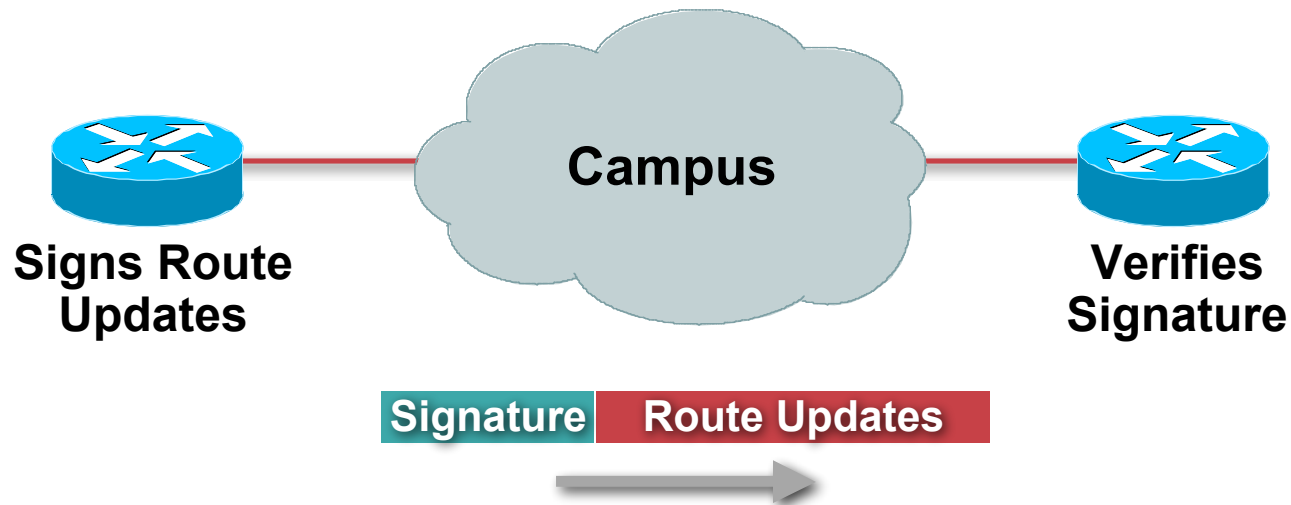
**Match anything of origin AS12163
and passed through AS1849**

MD5 Authentication of the Routing Protocol Updates

Secure Routing

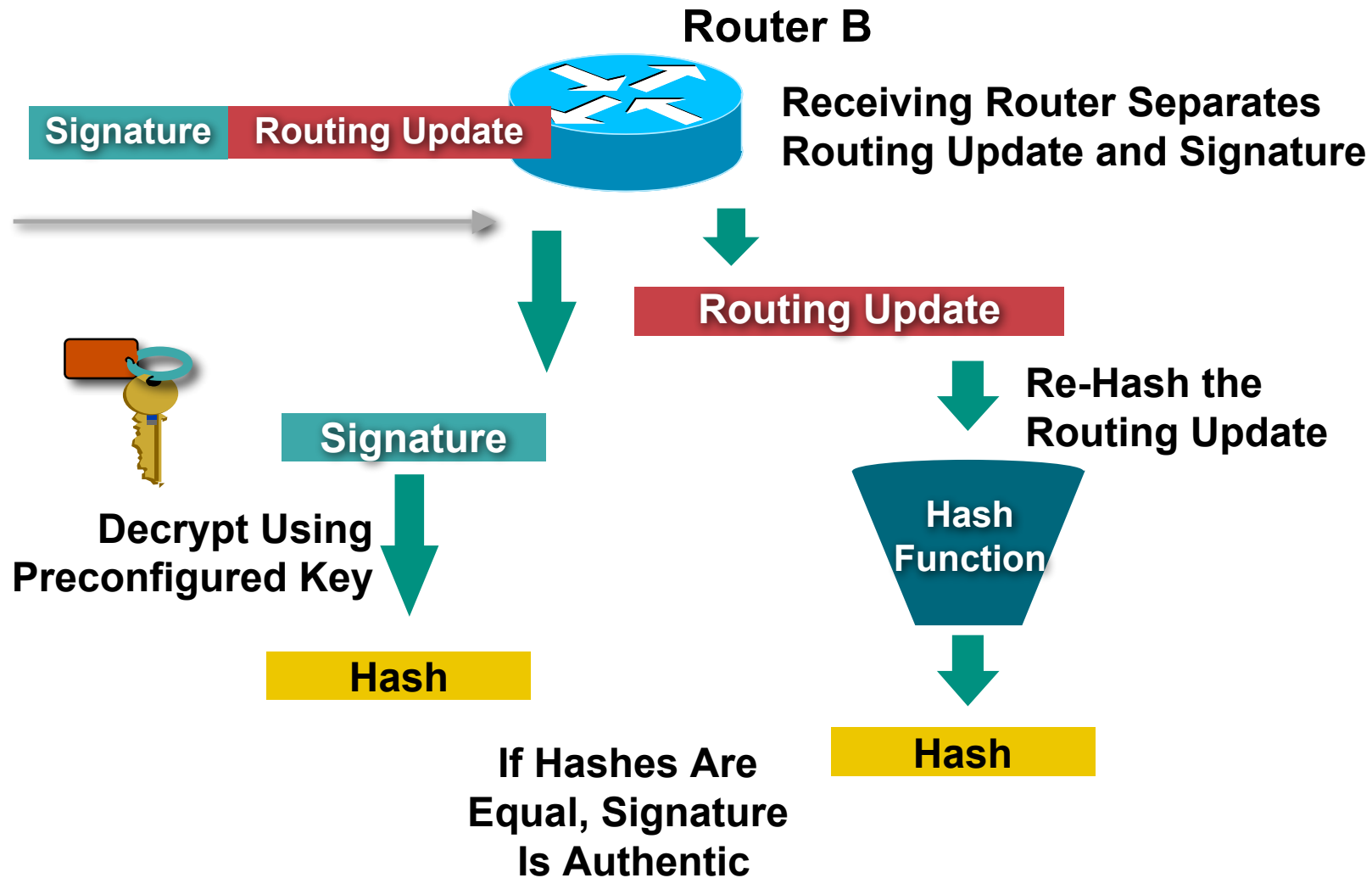
Route Authentication

Configure Routing Authentication



Certifies **Authenticity** of Neighbor
and **Integrity** of Route Updates

Signature Verification



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 cisco
!
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 cisco 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 cisco
```

Routing Protocol Convergence Speed and Security

Tune for Fast Convergence

- **Faster Convergence means the network can recover from a security incident.**

Interface Settings:

Increase the interface "hold-queue 1500" for each interface (default is 75). Do check your Line Card/VIP memory before increasing the hold-queue.

TCP Settings (improves BGP convergence):

ip tcp selective-ack

ip tcp mss 1460

ip tcp window-size 65535

ip tcp queuemax 50

ip tcp path-mtu-discovery

BGP BCPs That Help Build Security Resistance

BGP Maximum Prefix Tracking

- **Allow configuration of the maximum number of prefixes a BGP router will receive from a peer**
- **Two level control**

Warning threshold: Log warning message

**Maximum: Tear down the BGP peering,
manual intervention required to restart**

BGP Maximum Prefix Tracking

```
n e i g h b o r < x . x . x . x > m a x i m u m - p r e f i x < m a x > [ < t h r e s h o l d > ]  
[ w a r n i n g - o n l y ]
```

- **Threshold is an optional parameter between 1 to 100 percent**

Specify the percentage of <max> that a warning message will be generated; Default is 75%

- **Warning-only is an optional keyword which allows log messages to be generated but peering session will not be torn down**

Default Routes, ISPs, and Security

Avoid Default Routes

- **ISPs with full BGP feeds should avoid default routes.**
- **DOS/DDOS attack use spoofed addresses from the un-allocated IPV4 space.**

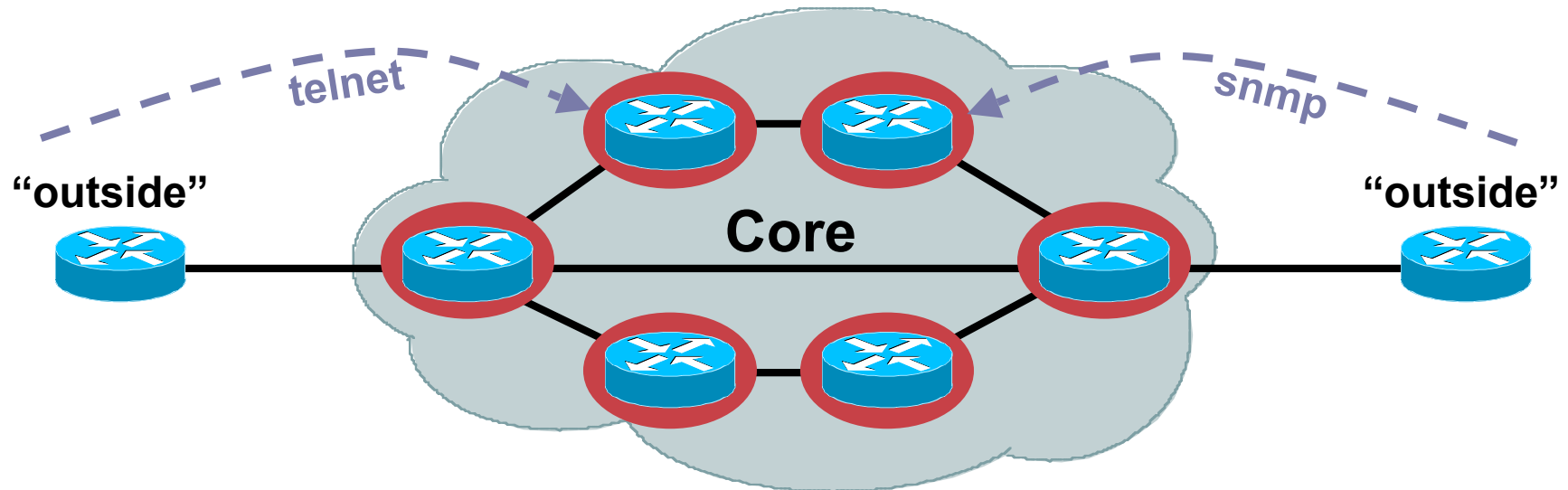
See <http://www.iana.org/assignments/ipv4-address-space> for the latest macro allocations.

- **Backscatter traffic from DOS/DDOS targets need to go somewhere. If there is a default, then this traffic will go to this one router and get dropped.**
- **Dropping backscatter traffic might overload the router.**

Infrastructure Security

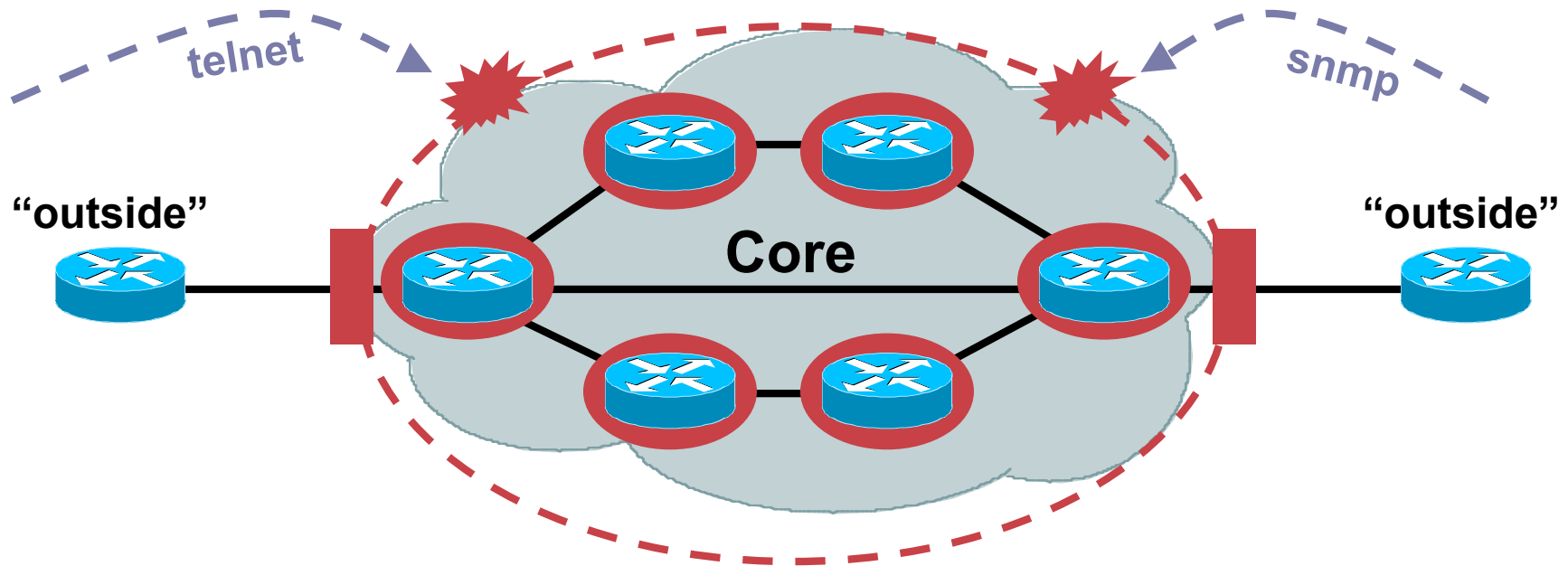


The Old World



- 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 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)**

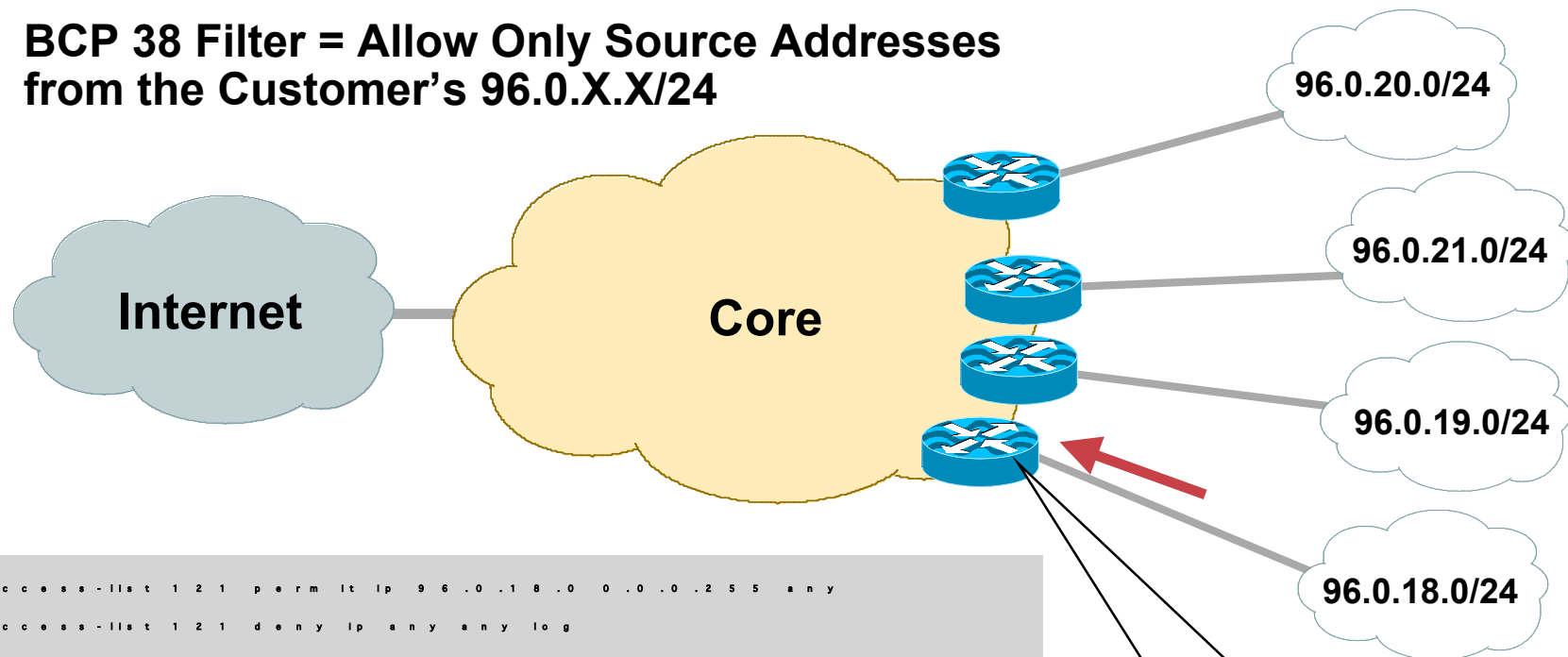
Using ACLs to Enforce BCP38

- **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

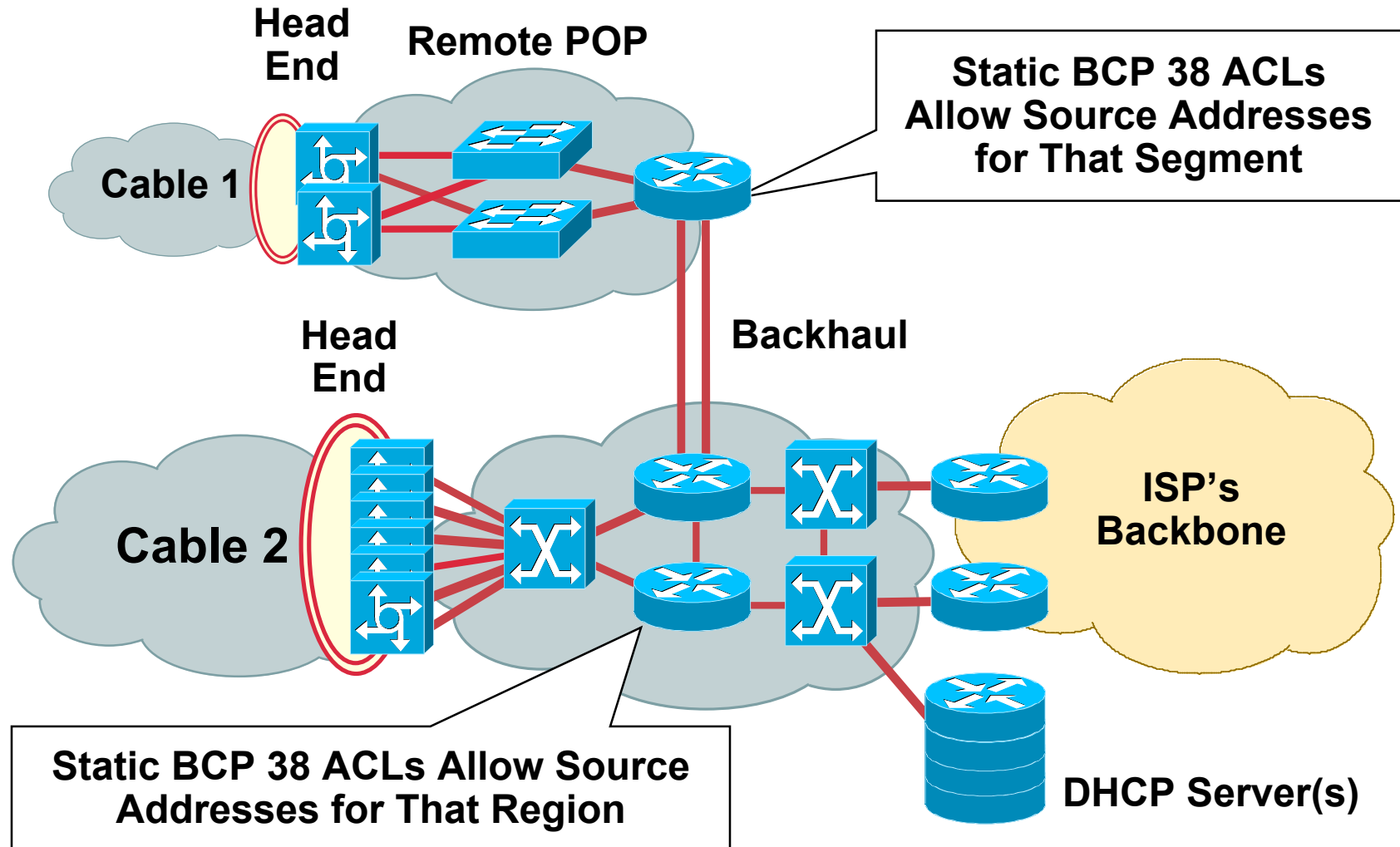


```
access-list 121 permit ip 96.0.18.0 0.0.0.255 any
access-list 121 deny ip any any log
!
interface serial 1 / 1 / 1 .3
description T1 Link to XYZ.
ip access-group 121 in
!
```

**BCP 38 Filter Applied
on Leased Line
Aggregation Router**

ISP

Static BCP 38 ACLs: DHCP



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**

Phase 1 – Preparation for the Attack

Securing the Network and Data Plane

What is Ingress and Egress?

Securing the Network

- **Route filtering**
- **Packet filtering**
- **Rate limits**

Techniques for BCP 38 Ingress Packet Filtering

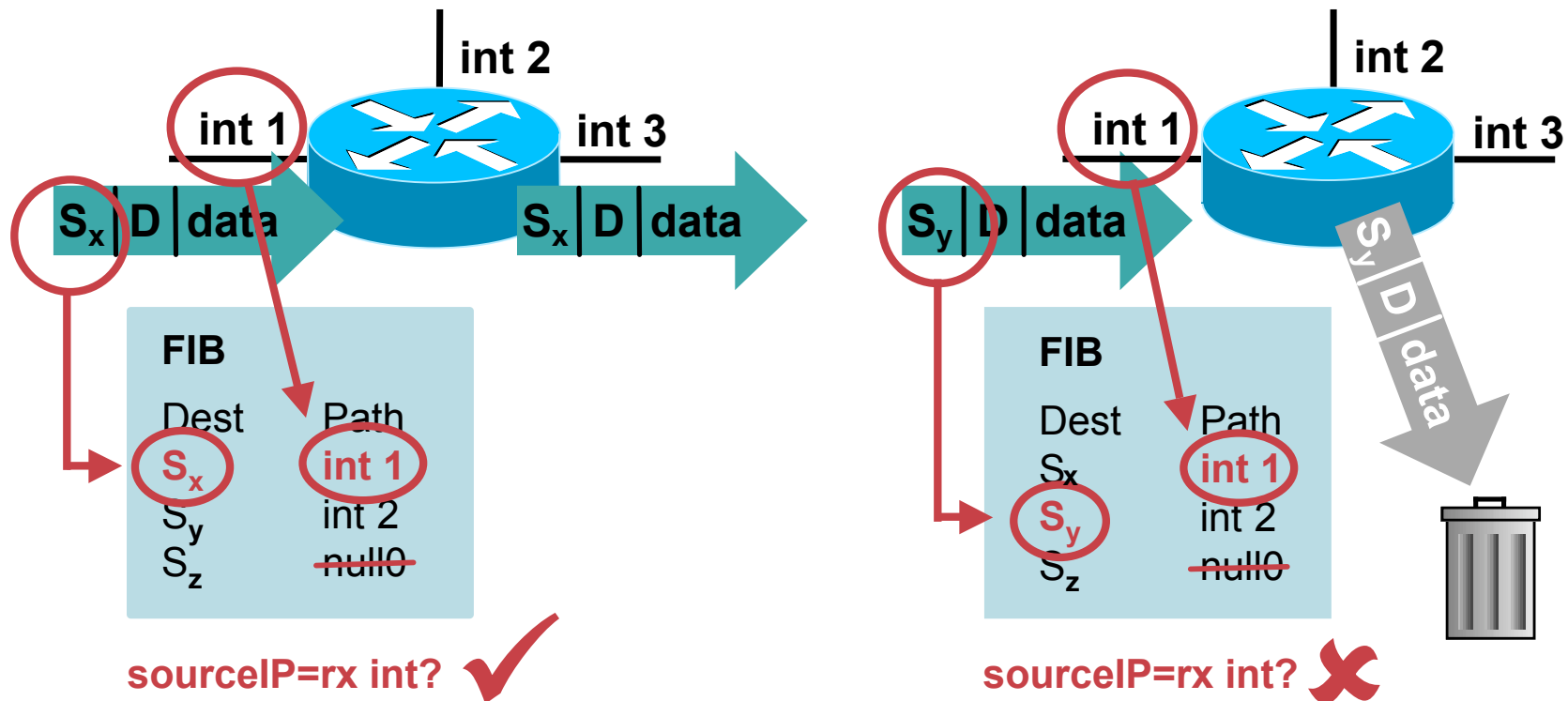
Strict Mode Unicast Reverse Path Forwarding (uRPF)

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

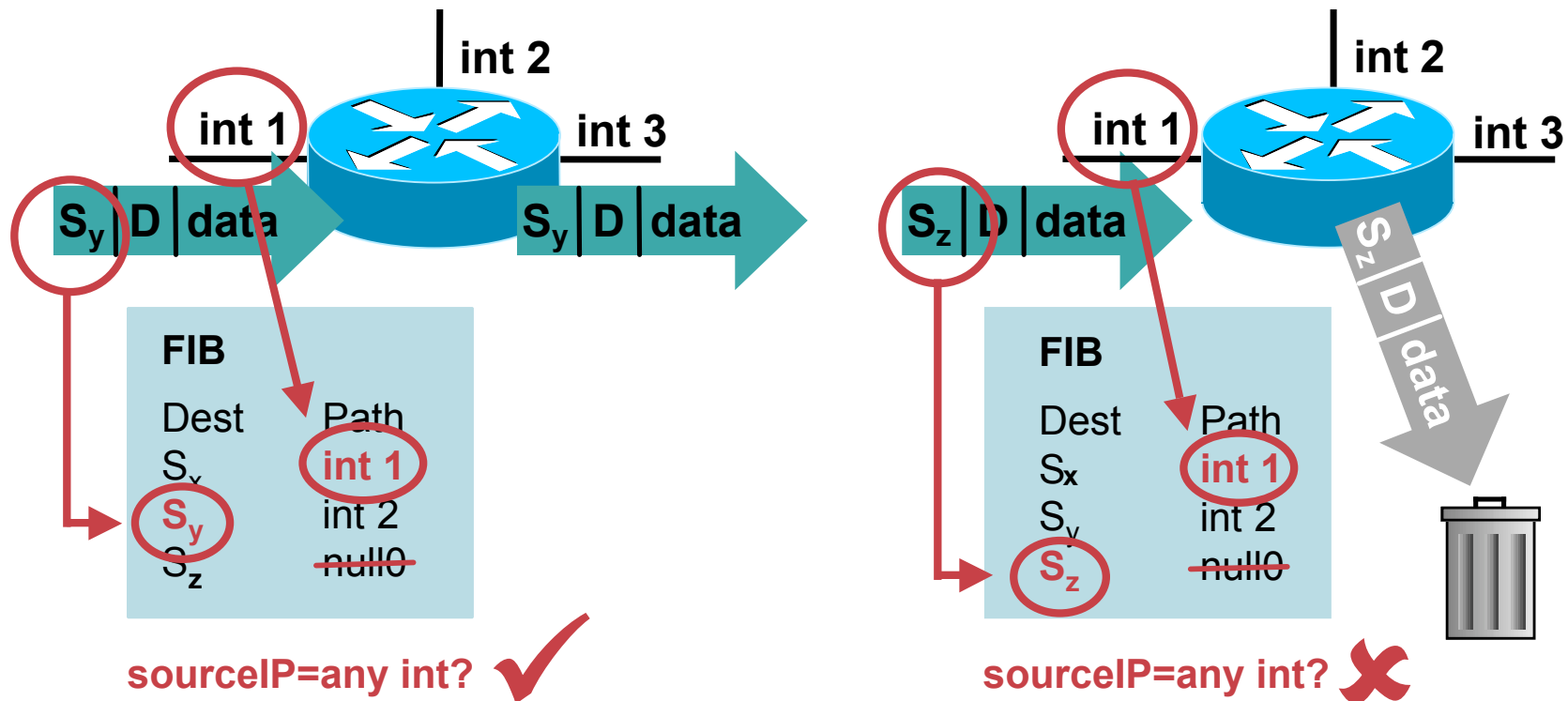
`router(config-if)# ip verify unicast source reachable-via rx`
(deprecated syntax: `ip verify unicast reverse-path`)



IP Verify Unicast Source Reachable—Via rx

uRPF—Loose Mode

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

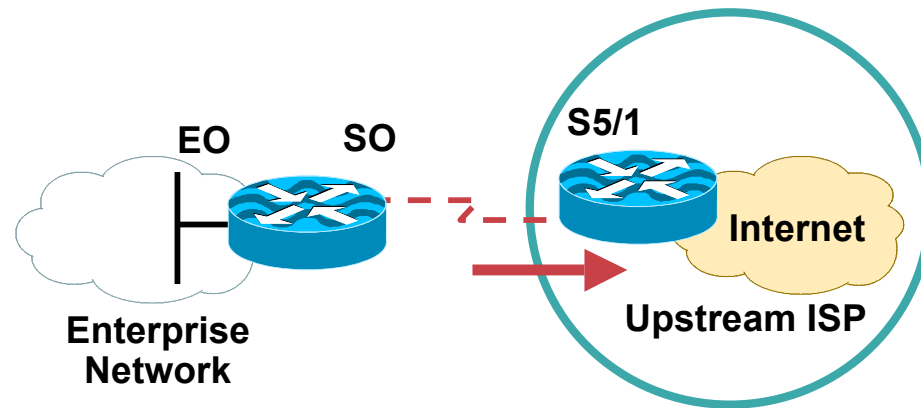


IP Verify Unicast Source Reachable—Via any

Unicast RPF (Strict Mode)

Simple Single Homed Customer Example

ISP Using uRPF for Ingress Filtering



```
interface Serial 5 / 0

description 1 2 8 K H D L C link to Galaxy Publications Ltd [galpub1] R5 - 0

bandwidth 128

ip unnumbered loopback 0

!Unicast RPF activated

ip verify unicast source reachable-via rx

no ip redirects

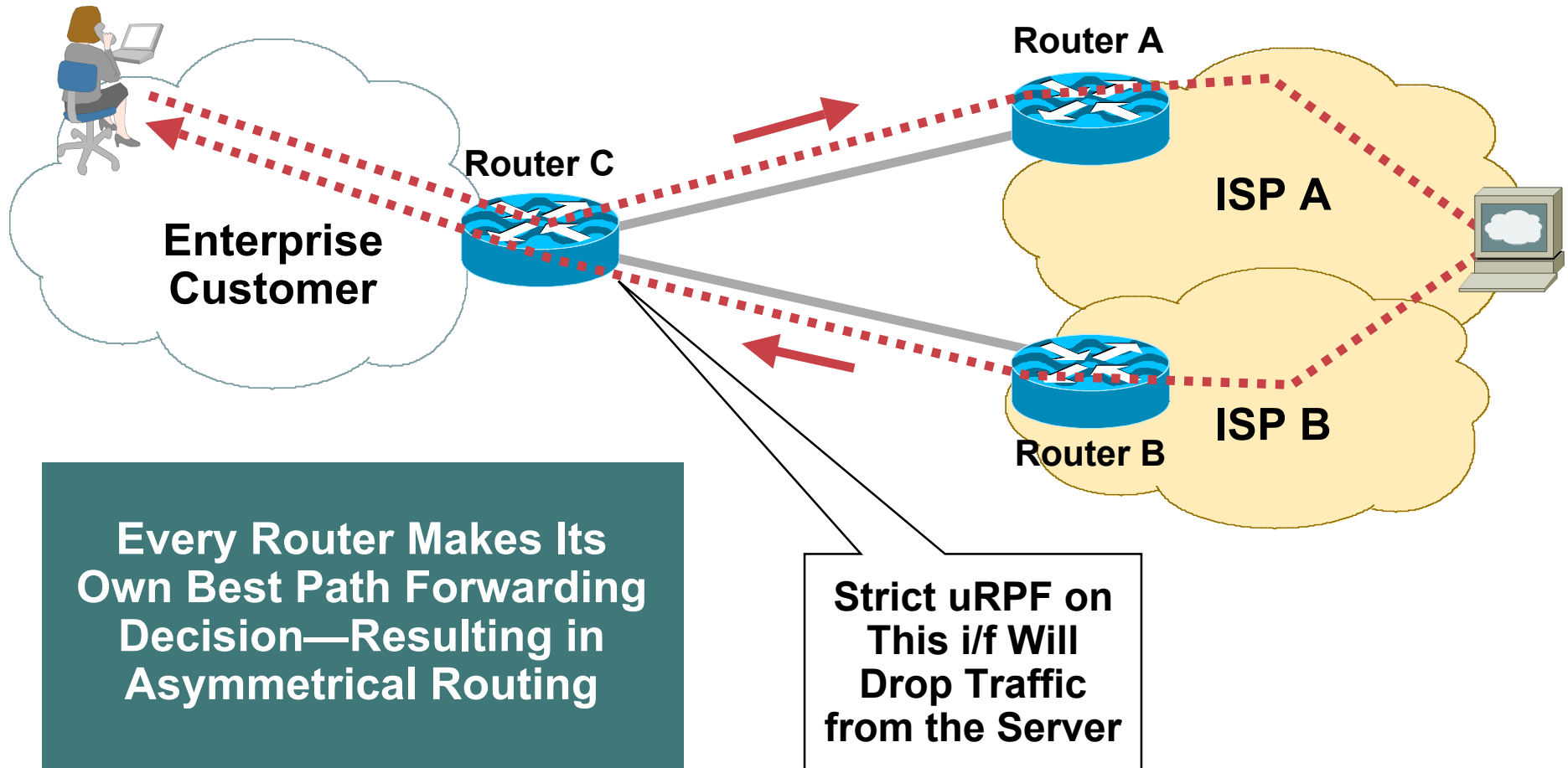
no ip directed-broadcast

no ip proxy-arp
```

!

uRPF and Multihomed Customers

What Is Asymmetrical Routing?



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

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**

Re-Coloring at the Edge – IP Precedence

What is Re-Coloring at the Edge?

- ***Re-Coloring*** is changing the precedence or DSCP part of the packet as it comes into your network.
- Precedence is used actively used on the Internet.

Routing Protocols are set to prec 6 and used with selective packet discard (SPD).

QOS/Diff-Serv used on in service providers.

Re-coloring at the Edge - IP Precedence

- **Some Internet sites change IP precedence so their content always “gets through”**

Recommended to reset IP precedence of incoming packets to default values (unless you know of traffic which needs different precedence values)

Running a Voice over IP network – inbound packets with highest precedence are “more important” than VoIP traffic, and will cause havoc in the local network

Some attacks set the prec/dscp values to give the attack an extra boost.

Phase 1 – Prepare the Tools and Techniques

Using IP Routing as a Security Tool

Using IP Routing as a Security Tool

- **IP Routing can be used to manipulate traffic on the ISPs network to:**

Null0 (Black Hole)

Shunts

Sink Hole

Analysis Devices

Clean up Devices

Rate-Limit

Using IP Routing as a Security Tool

- **Uses a BGP “trigger router”**

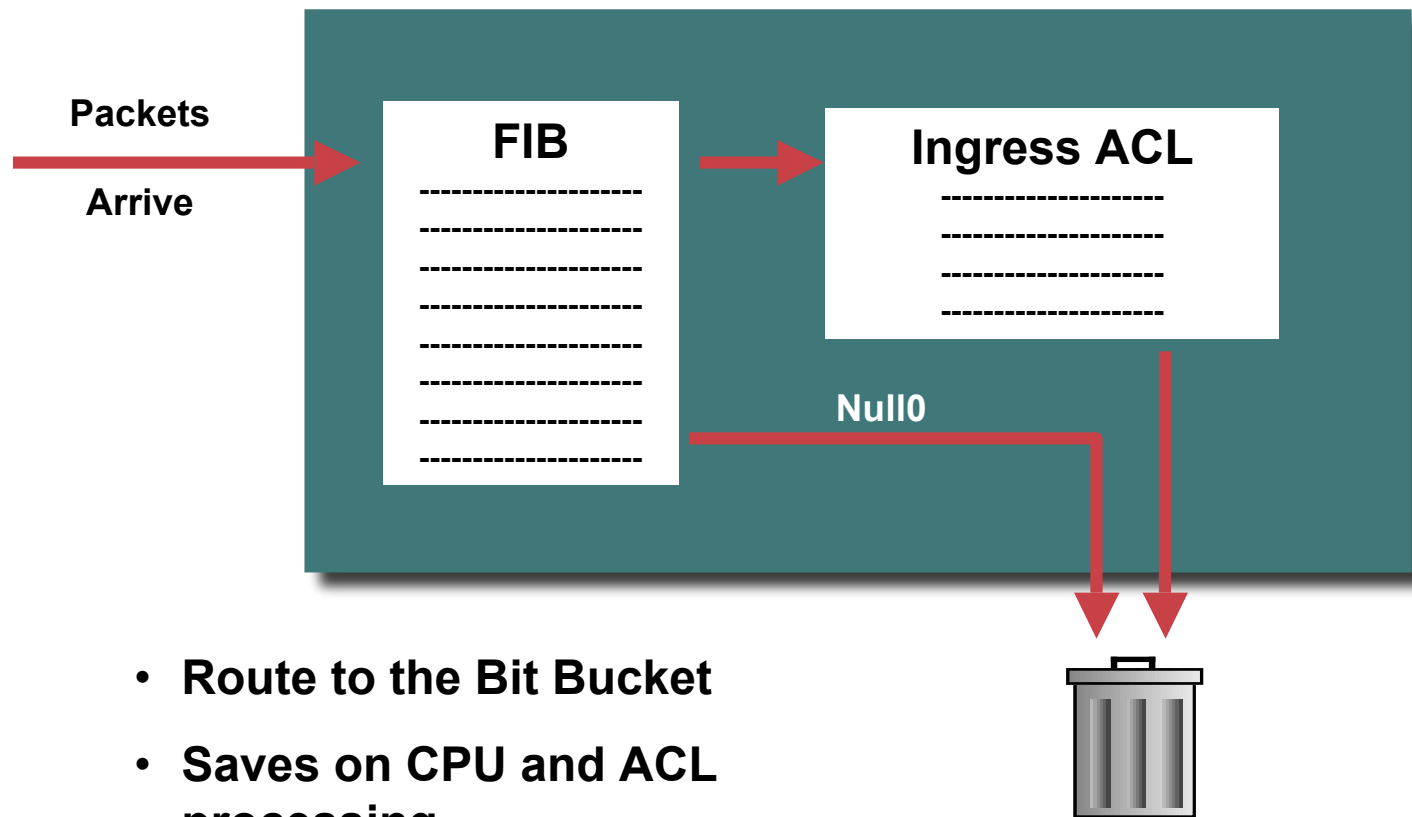
Phase 1 – Prepare the Tools and Techniques

Black Hole Filtering

Black Hole Filtering

- ***Black Hole Filtering or Black Hole 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 (depending on the forwarding ASIC).
- Used for years as a means to “black hole” unwanted packets.

Black Hole Filtering



- Route to the Bit Bucket
- Saves on CPU and ACL processing

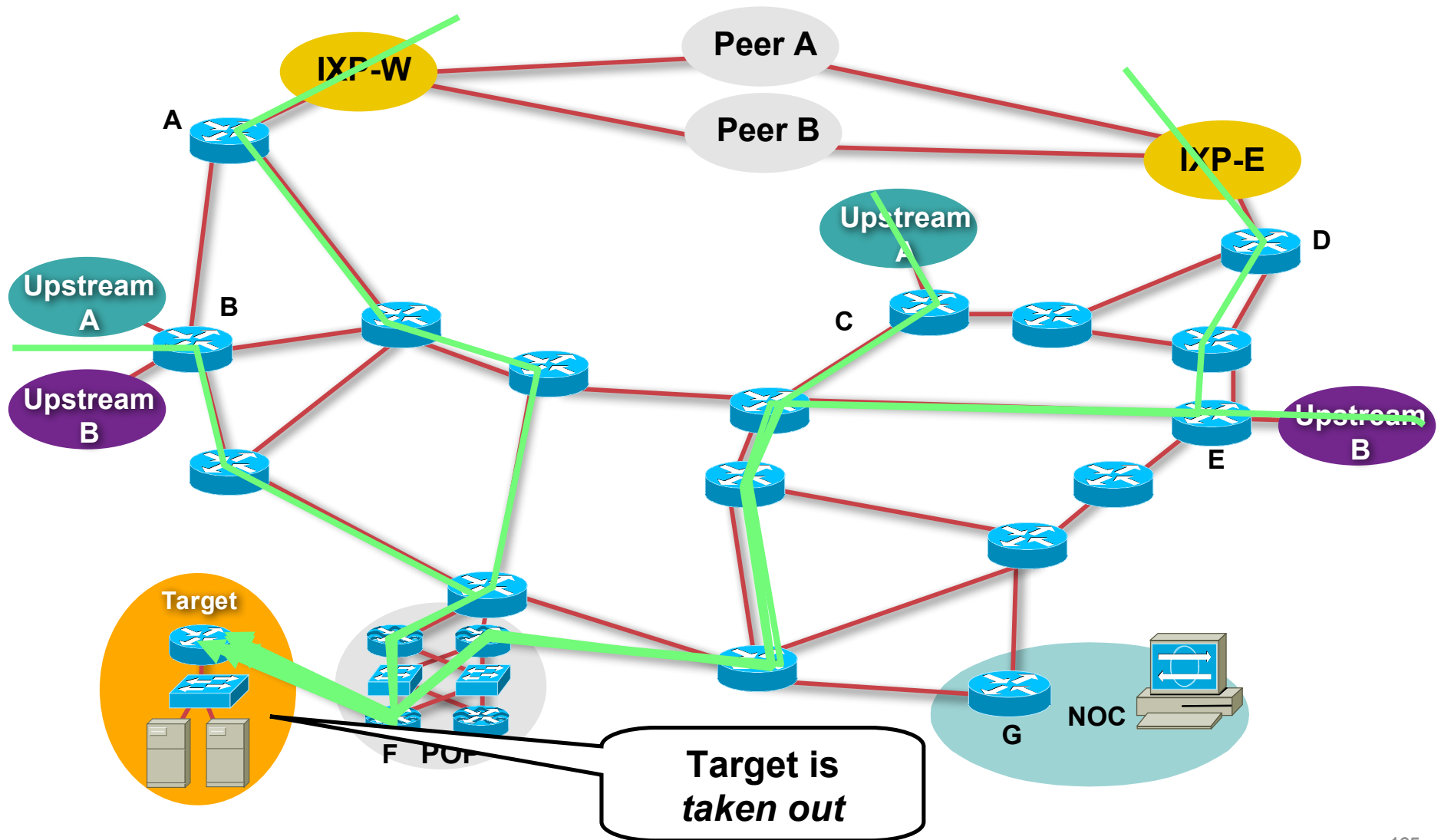
Phase 1 – Prepare the Tools and Techniques

Remote Triggered Black Hole Filtering

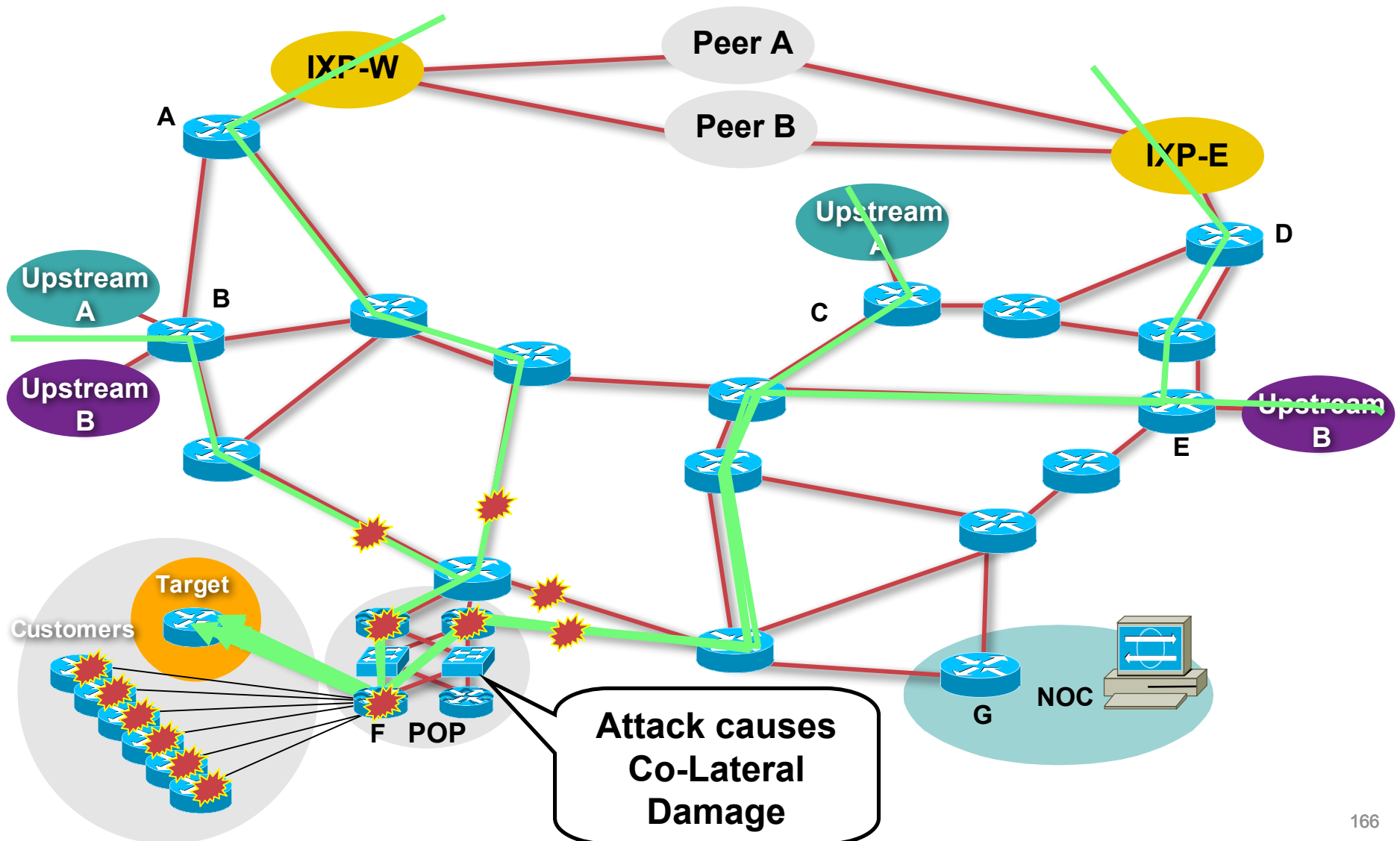
Remotely Triggered Black Hole Filtering

- **We use BGP to trigger a network wide response to an attack flow.**
- **A simple static route and BGP will allow an ISP to trigger network wide destination address black hole as fast as iBGP can update the network.**
- **This provides ISPs a tool that can be used to respond to security related events or used for DOS/DDOS Backscatter Tracebacks.**

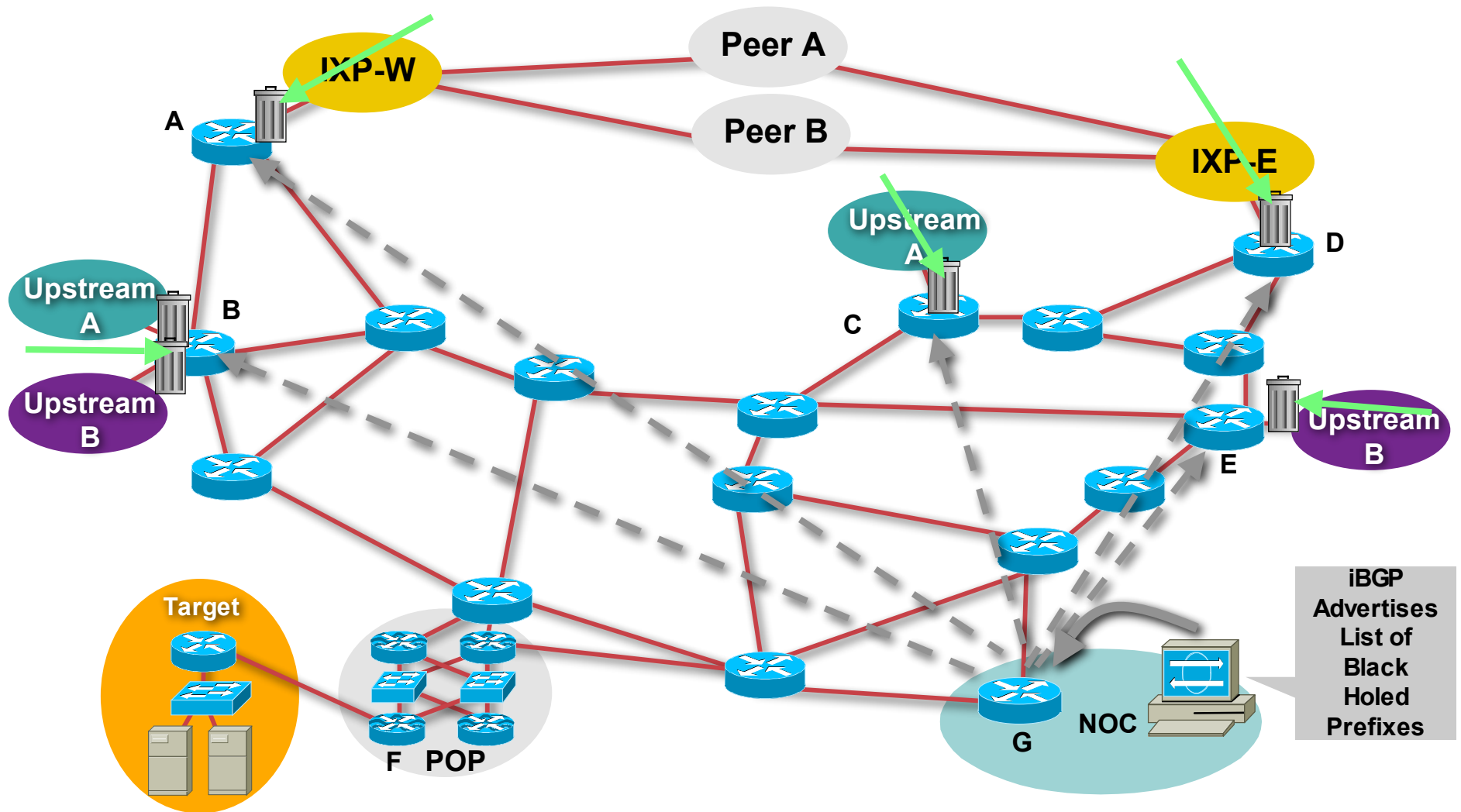
Customer is DOSed – Before



Customer is DOSed – Before – Co-Lateral Damage



Customer is DOSed – After – Packet Drops Pushed to the Edge



Remote Triggered Black Hole

- Remote Triggered Black Hole filtering is the foundation for a whole series of techniques to traceback and react to DOS/DDOS attacks on an ISP's network.
- Preparation does not effect ISP operations or performance.
- It does adds the option to an ISP's *security toolkit*.

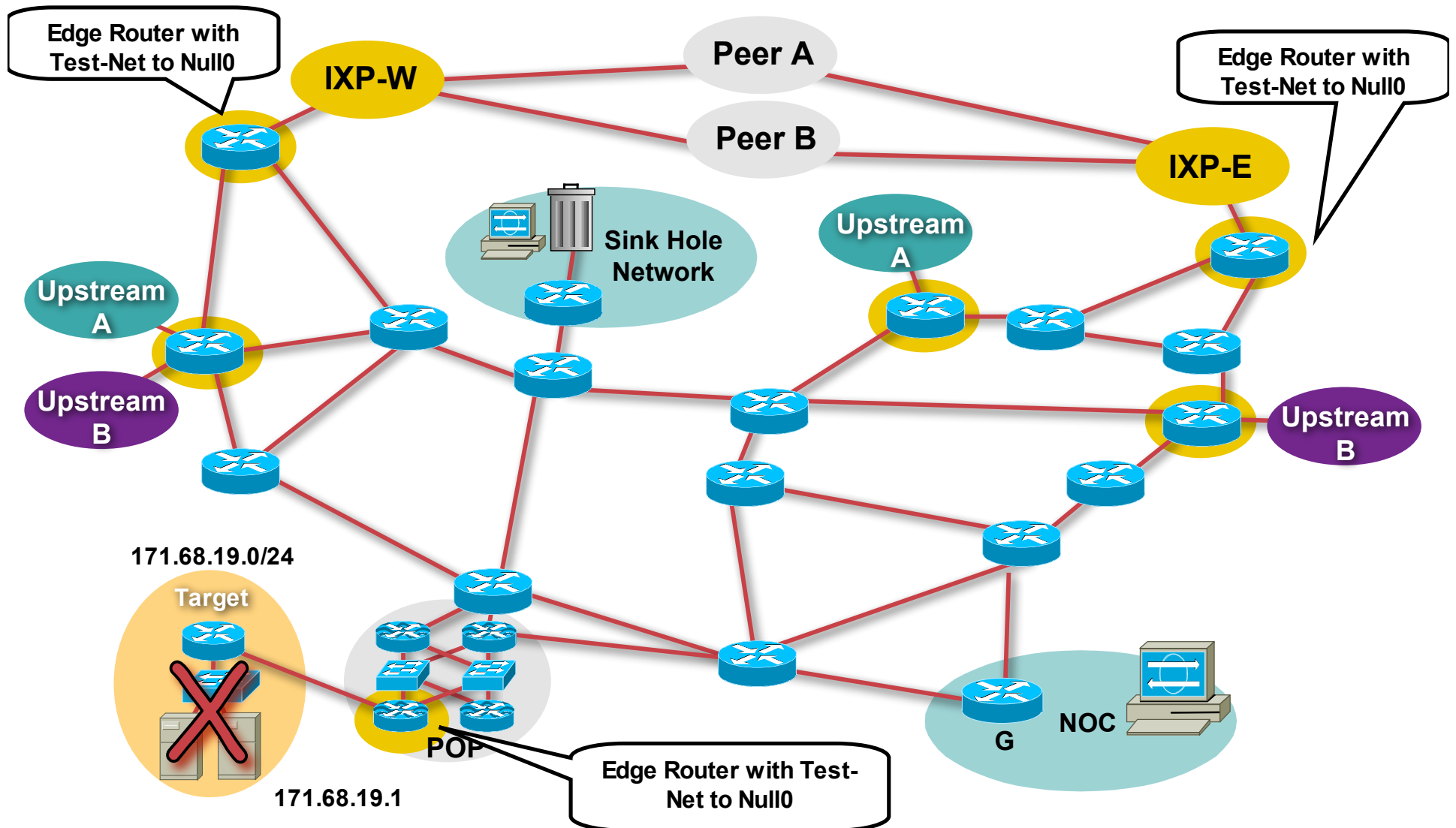
Step 1- Prepare all the Routers w/ Trigger

Select a small block that will not be used for anything other than black hole filtering. Test Net (192.0.2.0/24) is optimal since it should not be on the Net and is not really used.

Put a static route with Test Net – 192.0.2.0/24 to Null 0 on every router on the network.

```
ip route 192.0.2.1 255.255.255.255 Null0 255  
  
ip route 192.0.2.2 255.255.255.255 Null0 199  
  
ip route 192.0.2.3 255.255.255.255 Null0 50
```

Step 1- Prepare all the Routers w/ 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 (some ISPs do this)

Can be a workstation with Zebra (interface with Perl scripts and other tools).

Trigger Router's Config

Redistribute
Static with a
route-map

Match
Static
Route
Tag

Set Next-Hop
to the Trigger

Set Local-Pref

```
router bgp 109
.
 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 50

 set community no-export

 set origin igp

!

Route-map static-to-bgp permit 20
```


Step 3 – Activate the Black Hole

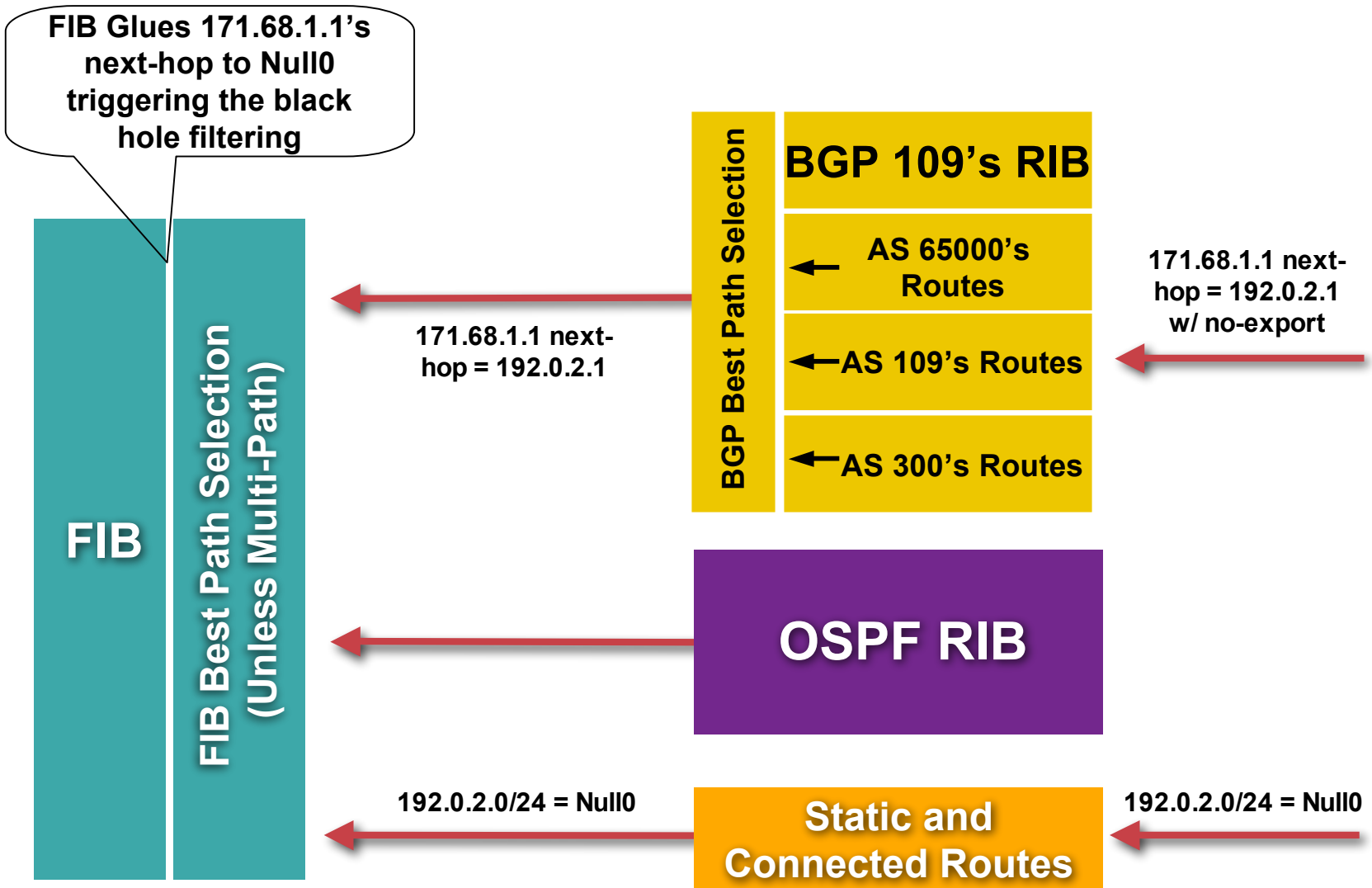
ISP adds a static route of the destination address they wish to black hole to the advertising router. The static is added with the “tag 66” to keep it separate from other statics on the router.

```
i p   r o u t e   1 7 1 . 6 8 . 1 . 1   2 5 5 . 2 5 5 . 2 5 5 . 2 5 5   N u l l 0   T a g   6 6
```

BGP Advertisement goes out to all BGP speaking routers.

Router hear the announcement, glues it to the existing static on the route, and changes the next-hop for the BGP advertised route to Null0 – triggering black hole routing.

Step 3 – Activate the Black Hole



Step 3 – Activate the Black Hole

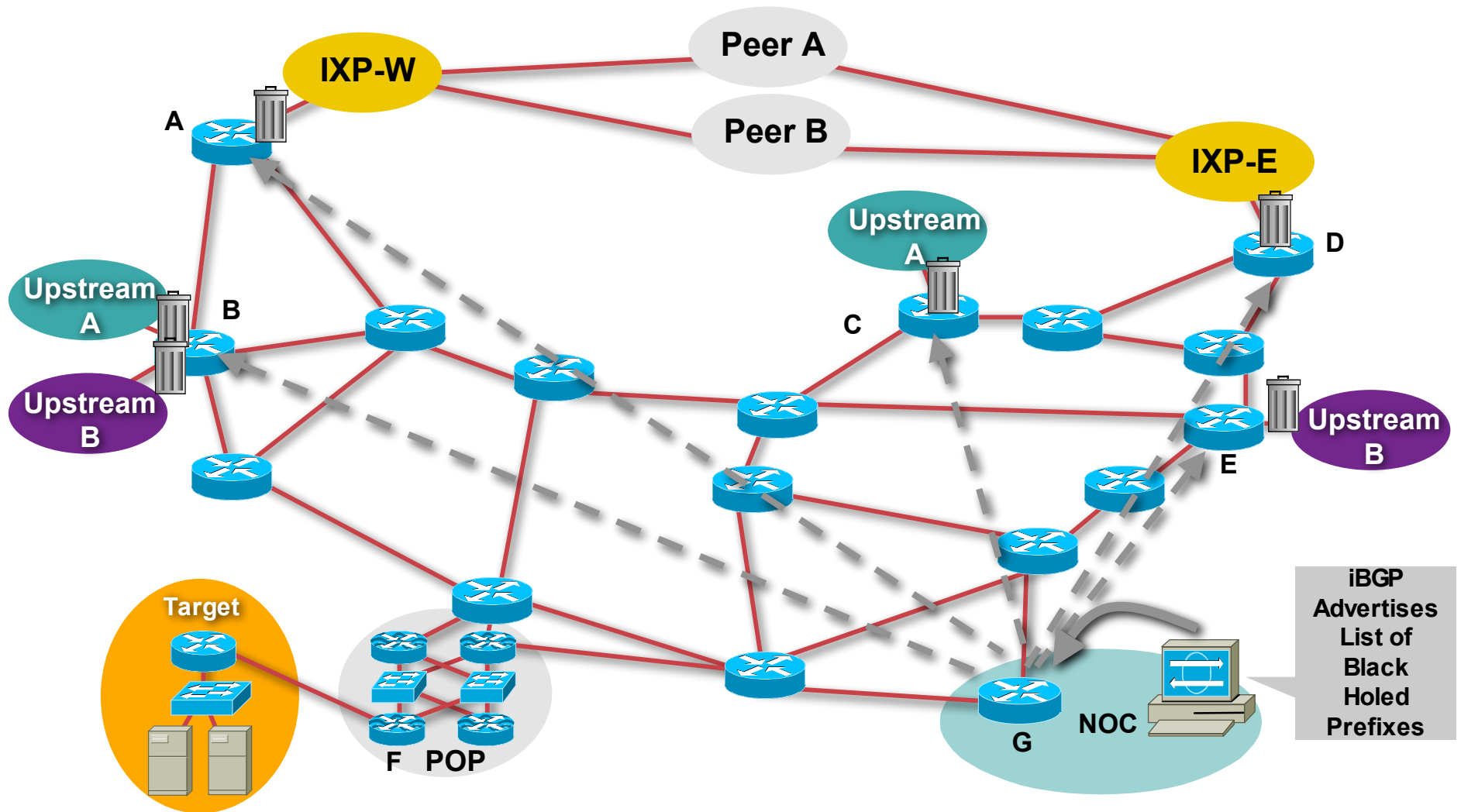
BGP Sent – 171.68.1.1 Next-Hop = 192.0.2.1

Static Route in Edge Router – 192.0.2.1 = Null0

171.68.1.1 = 192.0.2.1 = Null0

Next hop of 171.68.1.1 is now equal to Null0

Step 3 – Activate the Black Hole



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 Null 0 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 DOS/DDOS 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 black hole on one ISP Peer's connection. Allows for the DOSed customer to have partial service.

Gotchas with Black Hole Filtering

- **Routers were designed to forward traffic, not drop traffic.**
- **ASIC Based Forwarding can drop traffic at line rate.**
- **Processor Based Forwarding can have problems dropping large amounts of data.**
- **Remember the old shunt technique**

Phase 1 – Prepare the Tools and Techniques

Loose Check – Network Wide Sources Address Reaction Tool

The Requirement

- **What can we use?**

iBGP can be used to trigger a black hole (remote triggered black hole filtering).

Meets the 60 routers in 60 seconds requirement.

Black Hole Filtering (destination to Null0) works on all platforms.

Thousands of Null0s can be added to the FIB. Max black hole list = the max FIB size.

uRPF inspects the source address, checks the FIB + Adjacency, and pass/drop the packet.

Drops source addresses that = Null0, but would break on the ISP-ISP edge

The Requirement

- **Answer! Take out the Adjacency Check in uRPF!**

uRPF Loose Check – checks to see if the route exist in the FIB.

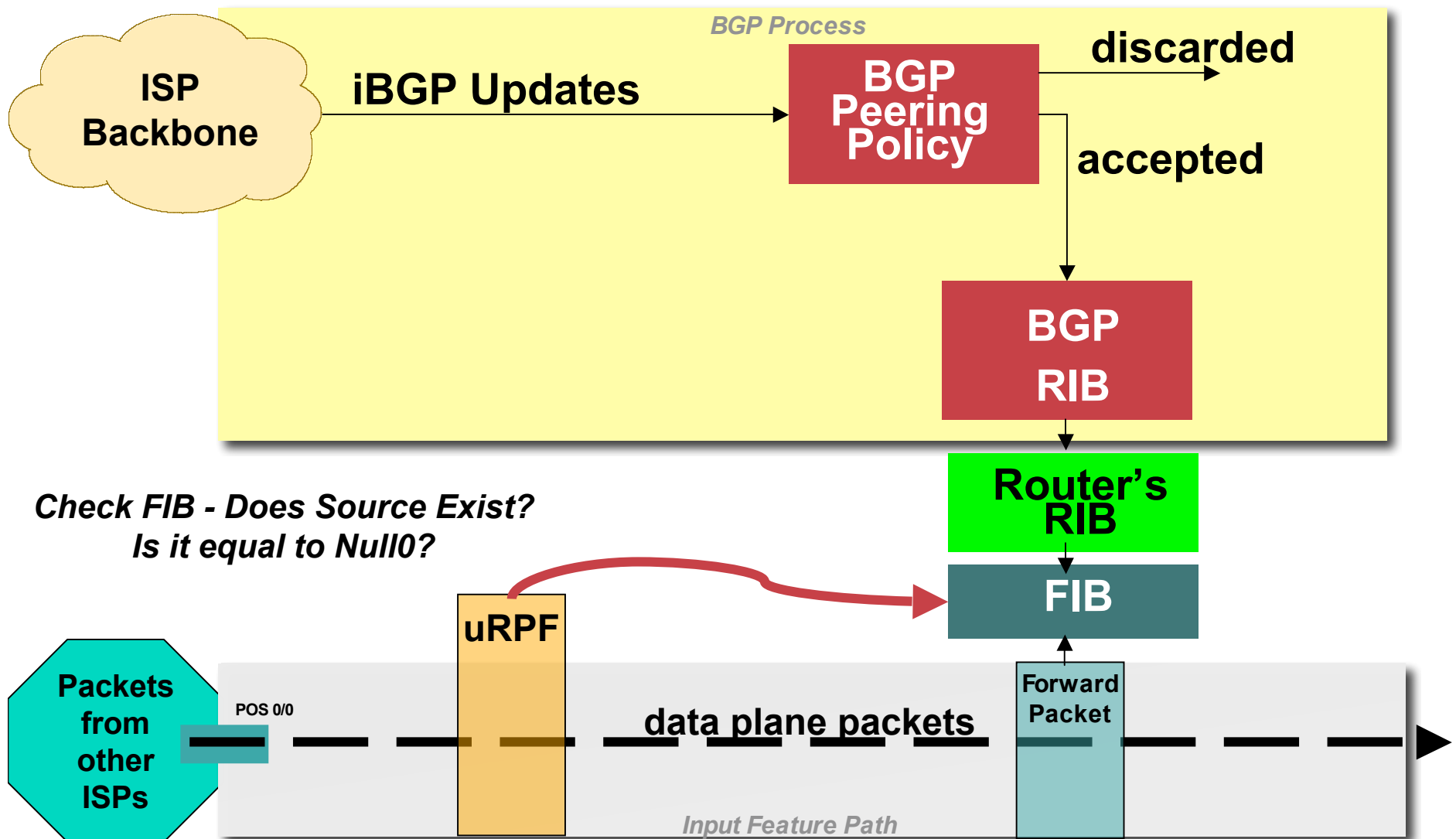
If not in FIB, drop the packet.

If equal to Null0, drop the packet.

No Adjacency Check – No problems with asymmetrical traffic flows on the ISP ISP edge.

Remote-Triggered Black Hole turned a prefix's next-hop to Null0 allowing uRPF Loose Check to drop the packets.

uRPF Loose Check



Source Based Remote Triggered Black Hole Filtering

- **What do we have?**

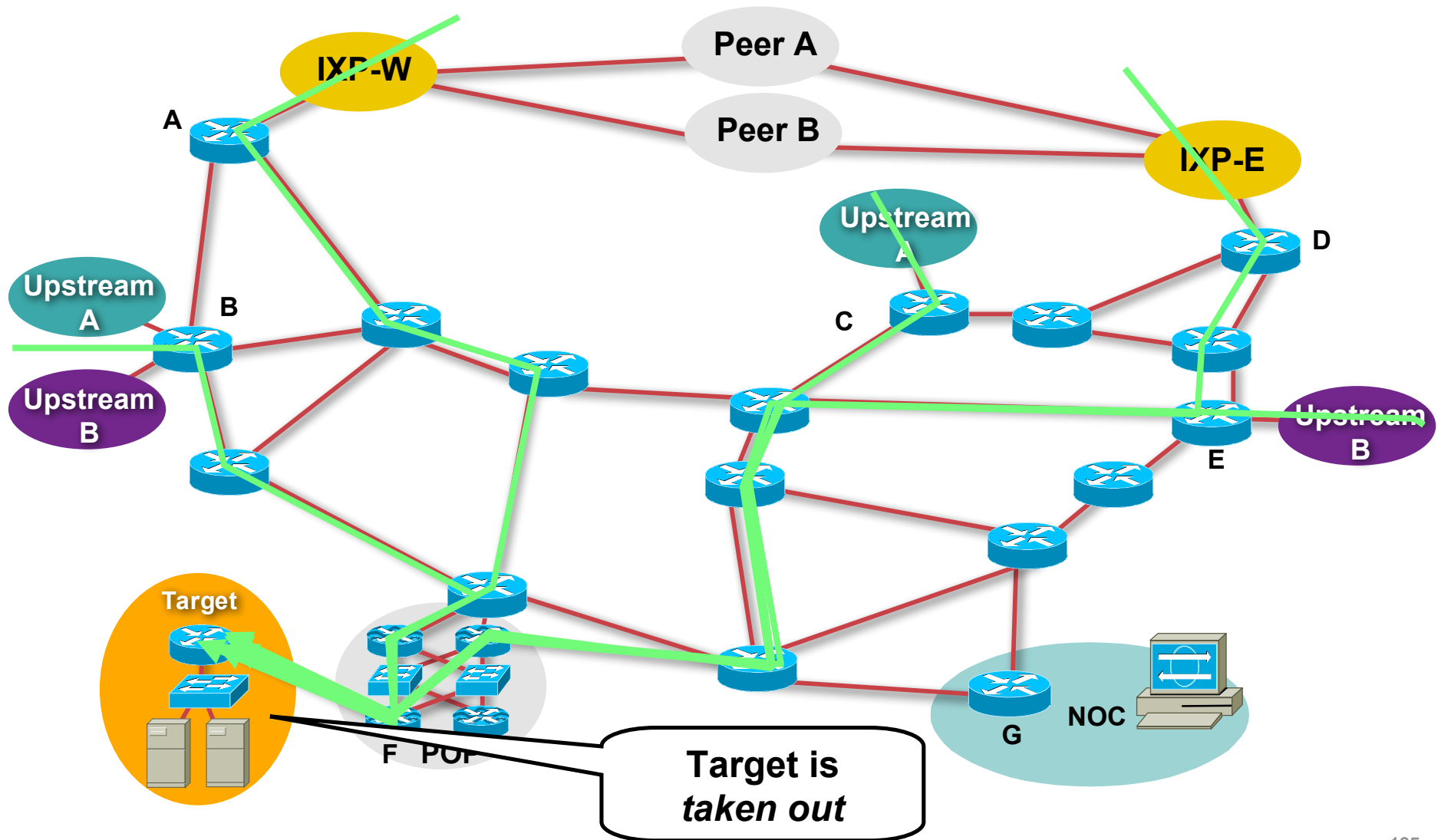
Black Hole Filtering – If the *destination* address equals Null 0 we drop the packet.

Remote Triggered – Trigger a prefix to equal Null 0 on routers across the Network at iBGP speeds.

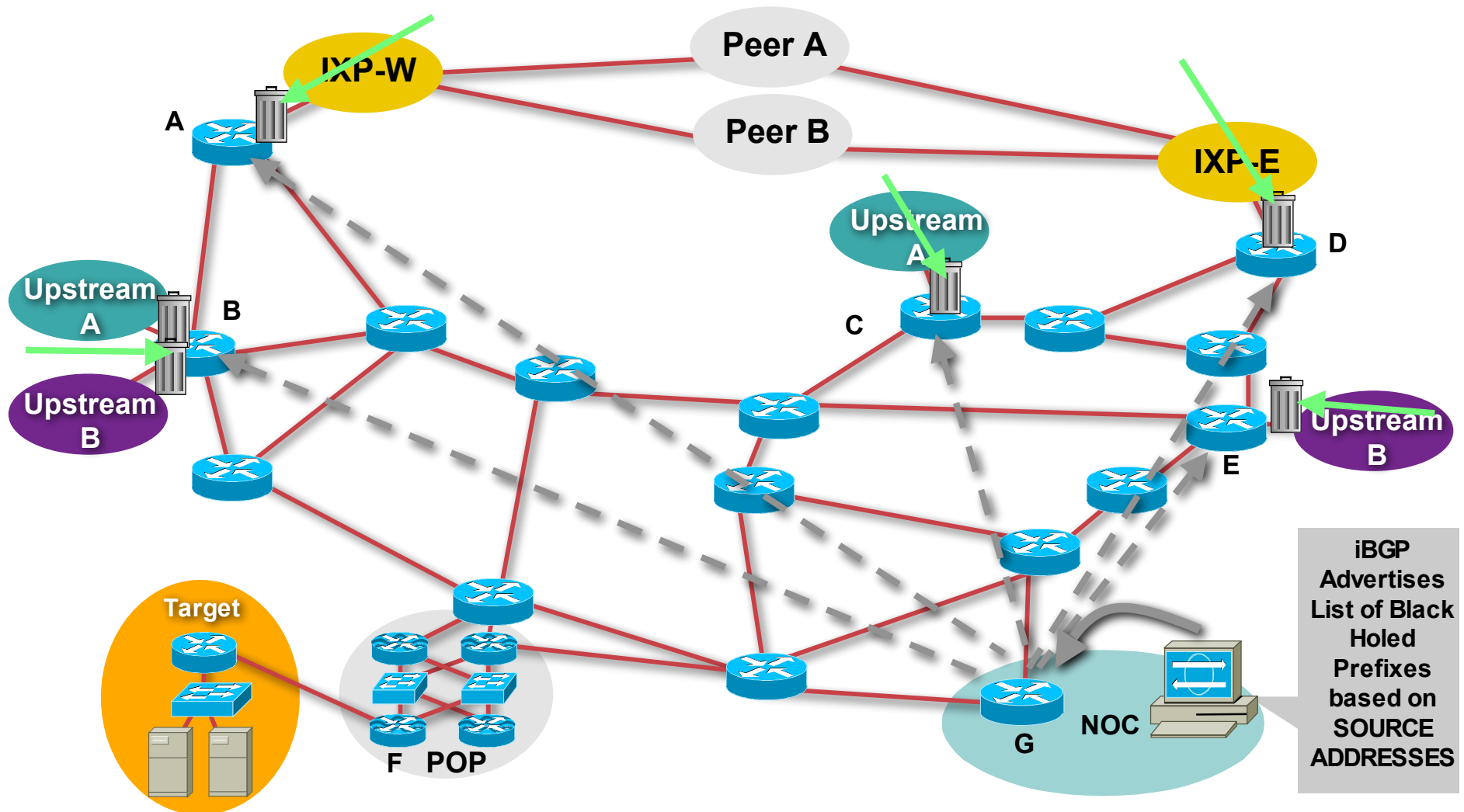
uRPF Loose Check – If the *source* address equals Null 0, we drop the packet.

- **Put them together and we have a tool to trigger drop for any packet coming into the network whose source or destination equals Null 0!**

Customer is DOSed - Before



Customer is DOSed – After



ACLs or uRPF Remote-Triggered Drop?

- **ACL's key strengths:**
 - Detailed Packet Filtering**
 - Static Filtering Environment**
 - Clear Filtering Policy**
- **ACL can have issues when faced:**
 - dynamic attack profiles (different sources, different entry points, etc)**
 - Frequent Changes**
 - Pushing out to multitude of devices**
- **Over-Lap ACLs with uRPF Remote-Triggered Drops allows for ACLs to handle the strict static policies while uRPF Remote-Triggered handles the dynamic source based drops.**

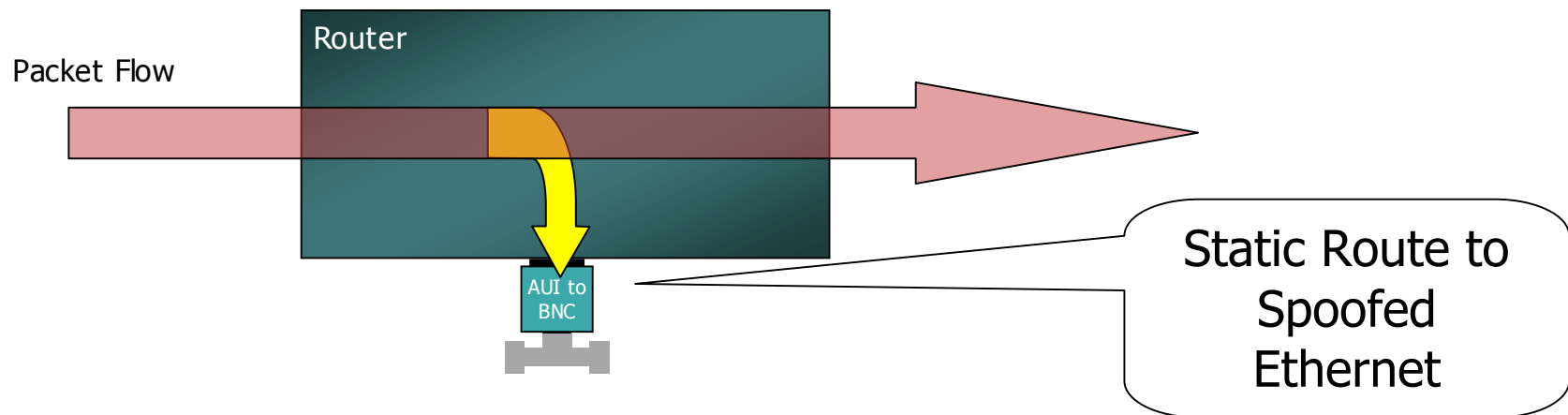
Phase 1 – Prepare the Tools and Techniques

Black Hole Shunts

Black Hole Shunt

- **Black Hole *Shunts*** are used to forward traffic out a spoofed interface.

Classic Example: AUI/BNC Transceiver with a T connector. A static MAC address is used with a static route.

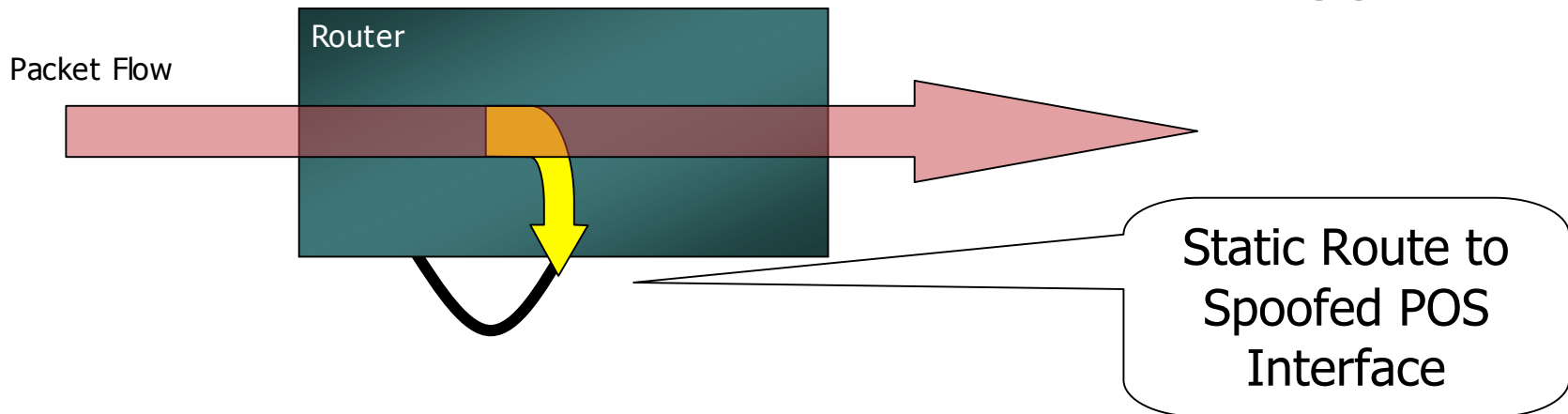


Remote Triggered Black Hole Shunt

- **BGP can be used to remote-trigger a next-hop redirect to a black hole shunt.**

A non-preferred BGP advertisement is sent with a “trigger community” and a next-hop to the “shunt.”

Local Route-Map used to prefer the trigger.



Trigger Router's Config

```
router bgp 109

.

 redistribute static route-map static-to-bgp

.

!

route-map static-to-bgp permit 20

 description - Redirect Traffic to Router's Shunt

 description - Use Static Route with Tag of 69

 match tag 69

 set local-preference 50

 set origin igp

 set community 600:9

!!

Route-map static-to-bgp permit 60
```

Edge Router with Shunt Interface

```
router bgp 109

.

neighbor INTERNAL route-map Counter-Measure-Select In

!

route-map Counter-Measure-Select permit 5

description - Set Next-Hop to Router's Shunt Interface

match community 199

set ip next-hop 60.40.1.2

set local-preference 500

!

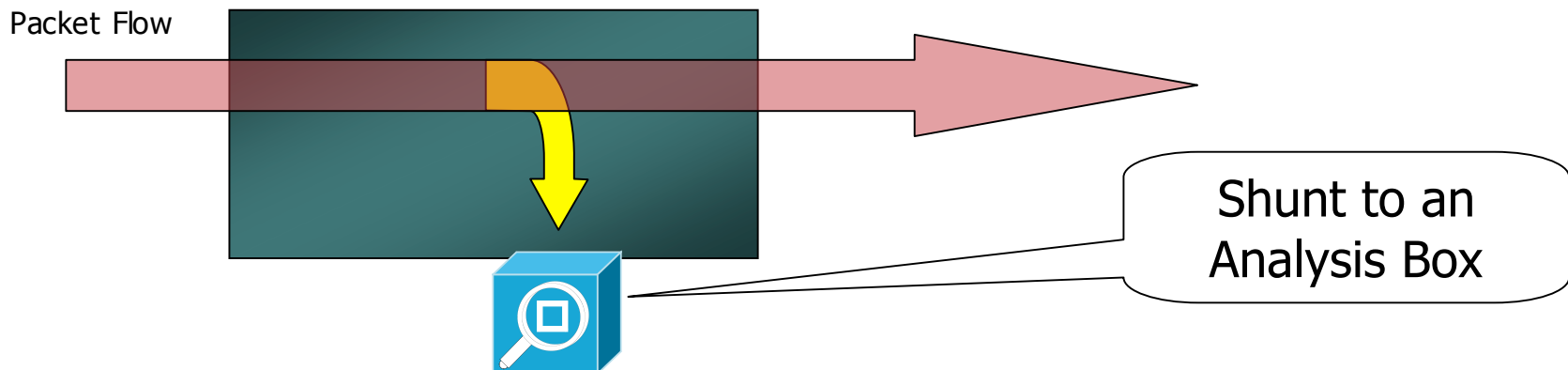
Route-map Counter-Measure-Select permit 60
```

Black Hole Shunt with Analysis

- **Black Hole Shunt interface can have a analysis tool on it.**

This allows for the dropped packets to be monitored, analyzed, and logged.

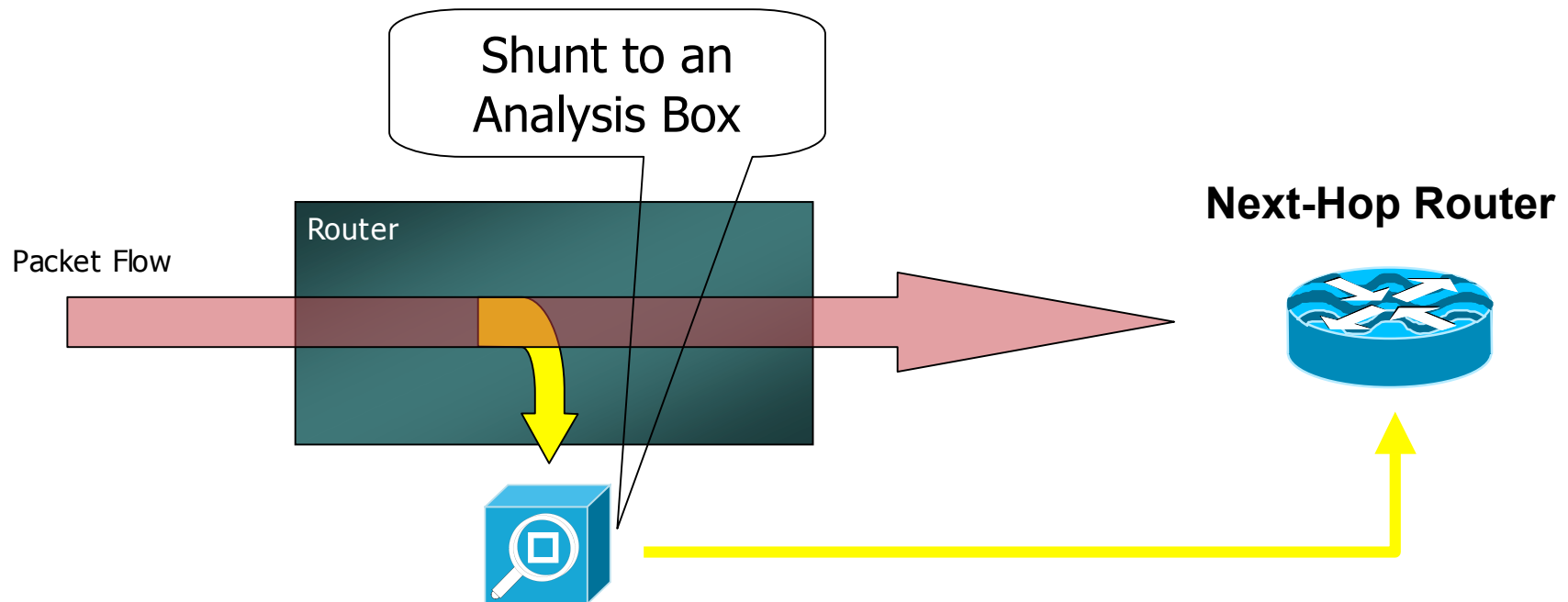
Sniffers, Commercial Tools, and Customized boxes are all options.



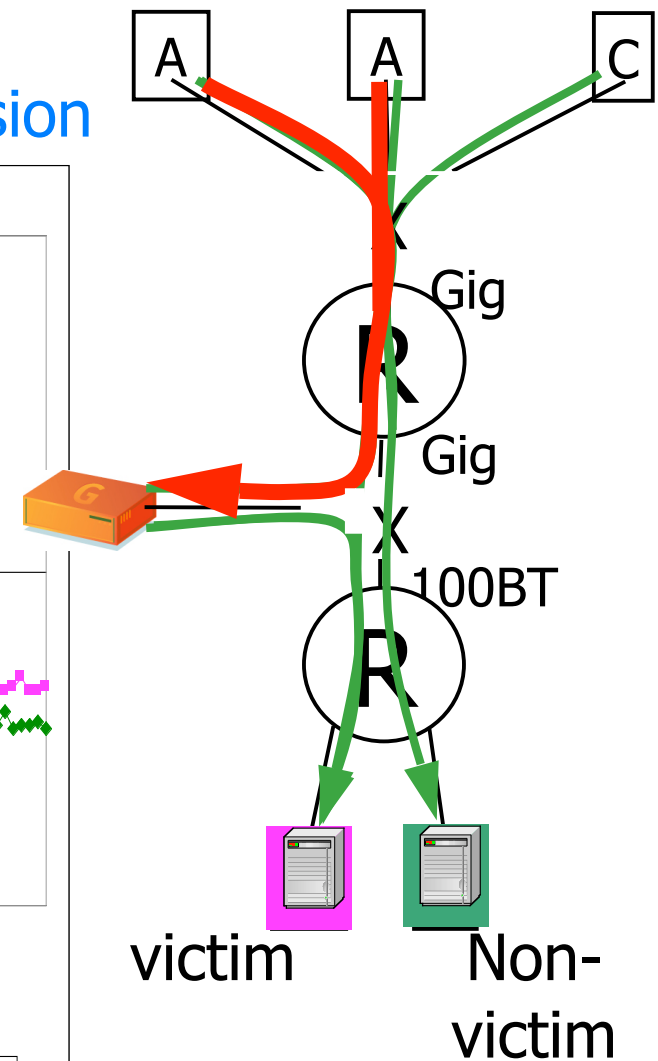
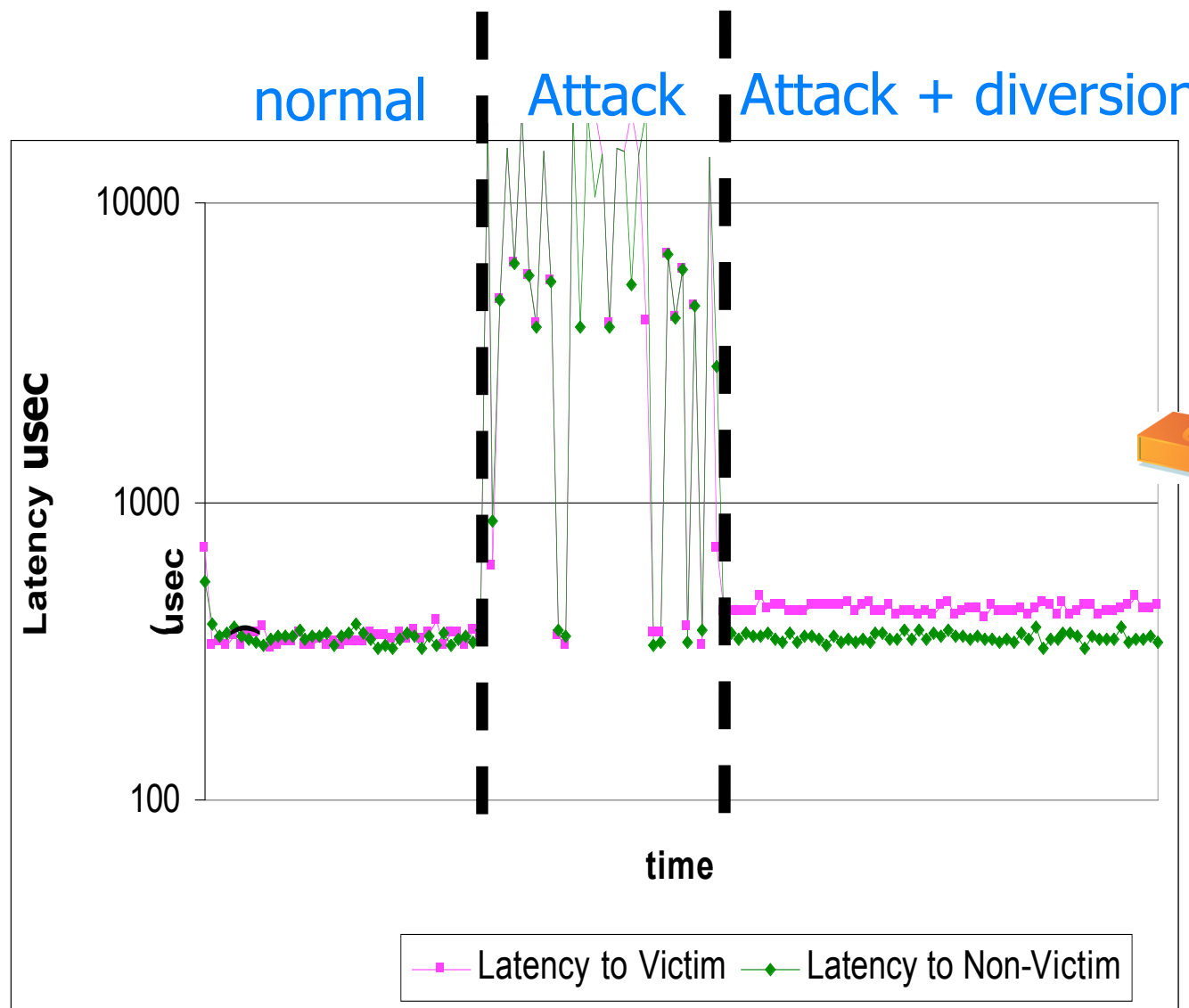
Black Hole Shunt with Clean-Up

- **Black Hole Shunt interface can have a clean up application (i.e. like Riverhead)**

This allows for the dropped packets to be analyzed, logged, cleaned up, and forwarded.



Black Hole Shunt with Clean-Up



Phase 1 – Prepare the Tools and Techniques

Sink Hole Routers/Networks

Sink Hole Routers/Networks

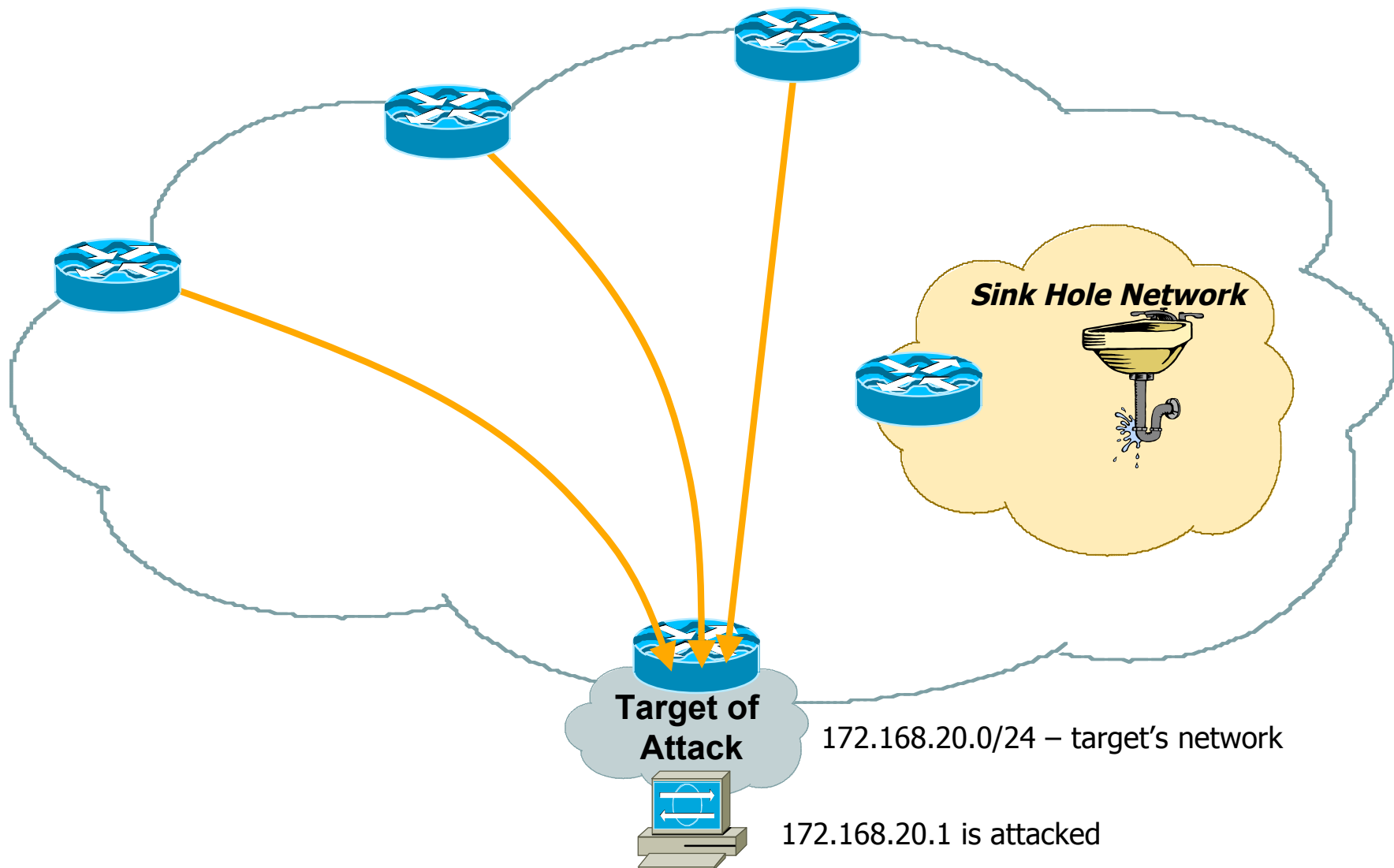
- Sink Holes are a the network equivalent of a honey pot.

BGP speaking Router or Workstation that built to *suck in* attacks.

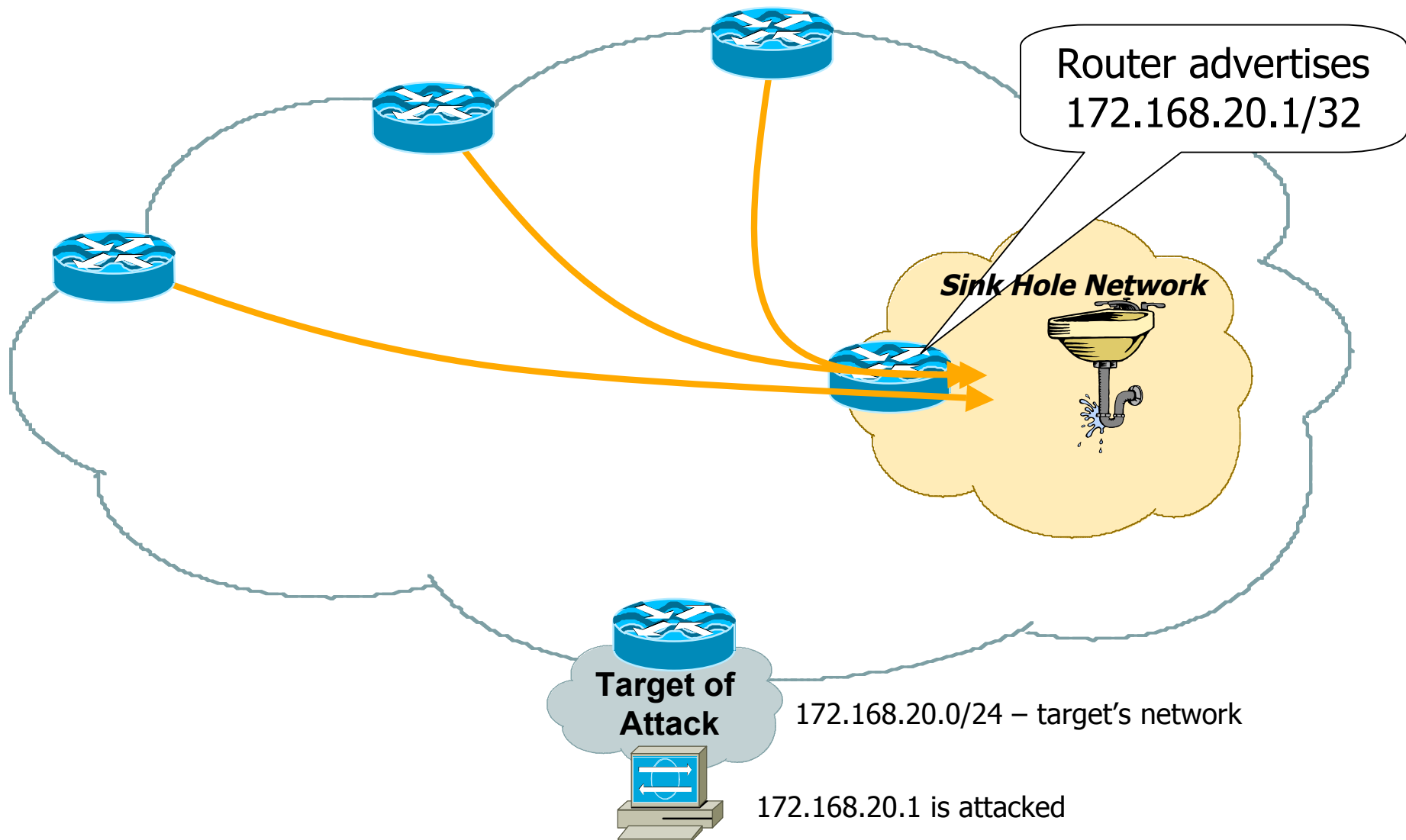
Used to redirect attacks away from the customer – working the attack on a router built to withstand the attack.

Used to monitor *attack noise, scans*, and other activity (via the advertisement of default)

Sink Hole Routers/Networks

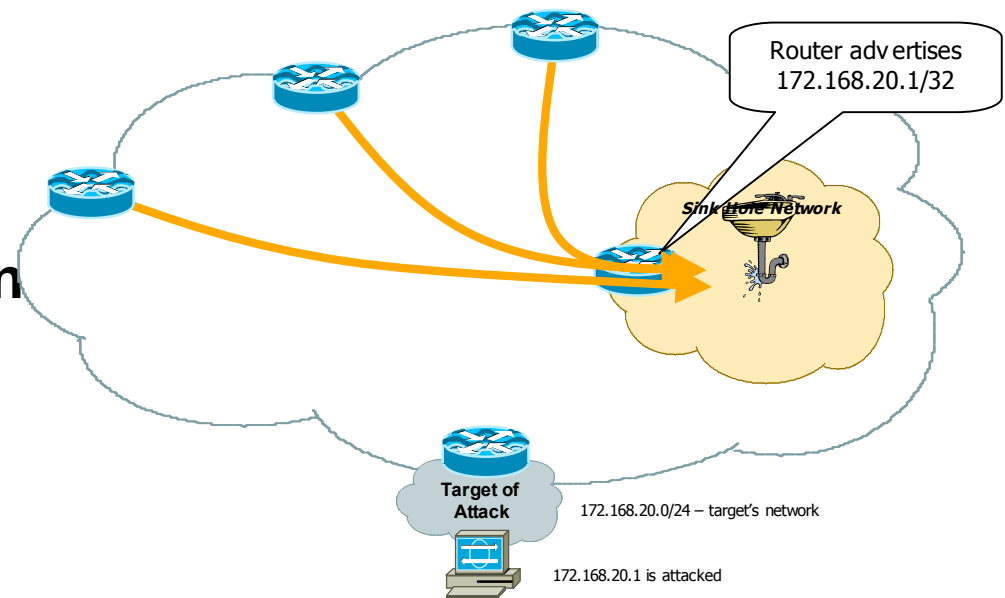


Sink Hole Routers/Networks



Sink Hole Routers/Networks

- **Attack is pulled off customer and your aggregation router.**
- **Can now do classification ACLs, Flow Analysis, Sniffer Capture, Traceback, etc.**
- **Objective is to minimize the risk to the network while working the attack incident.**



Sink Hole Routers/Networks

- Advertising Default from the Sink Hole will pull down all sort of *junk* traffic.

Customer Traffic when circuits flap.

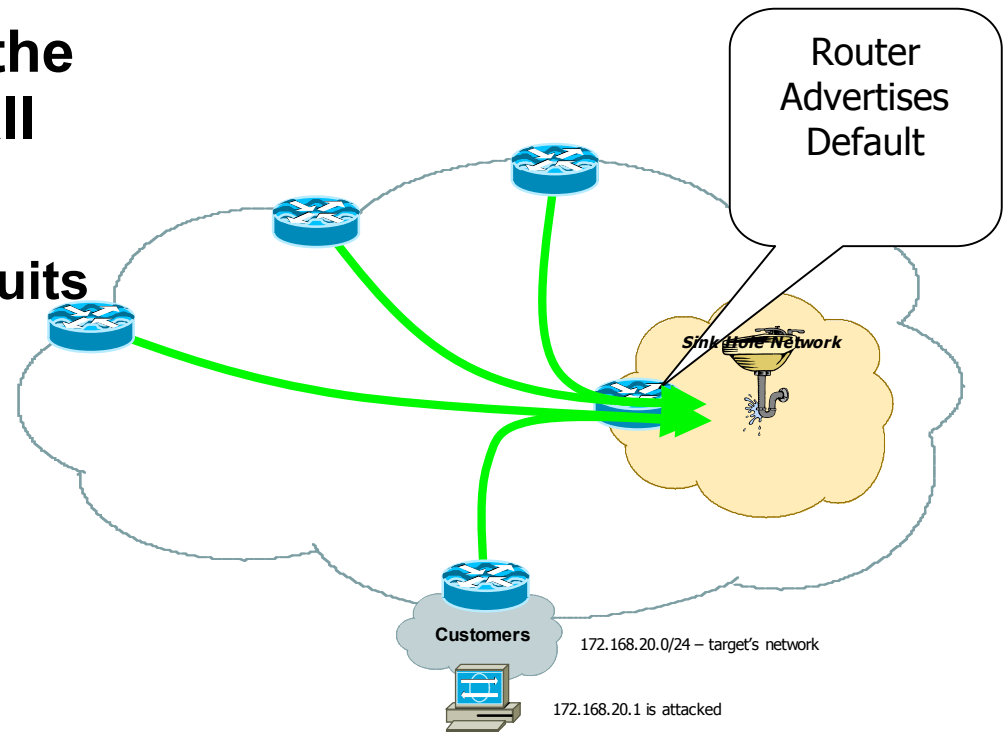
Network Scans

Failed Attacks

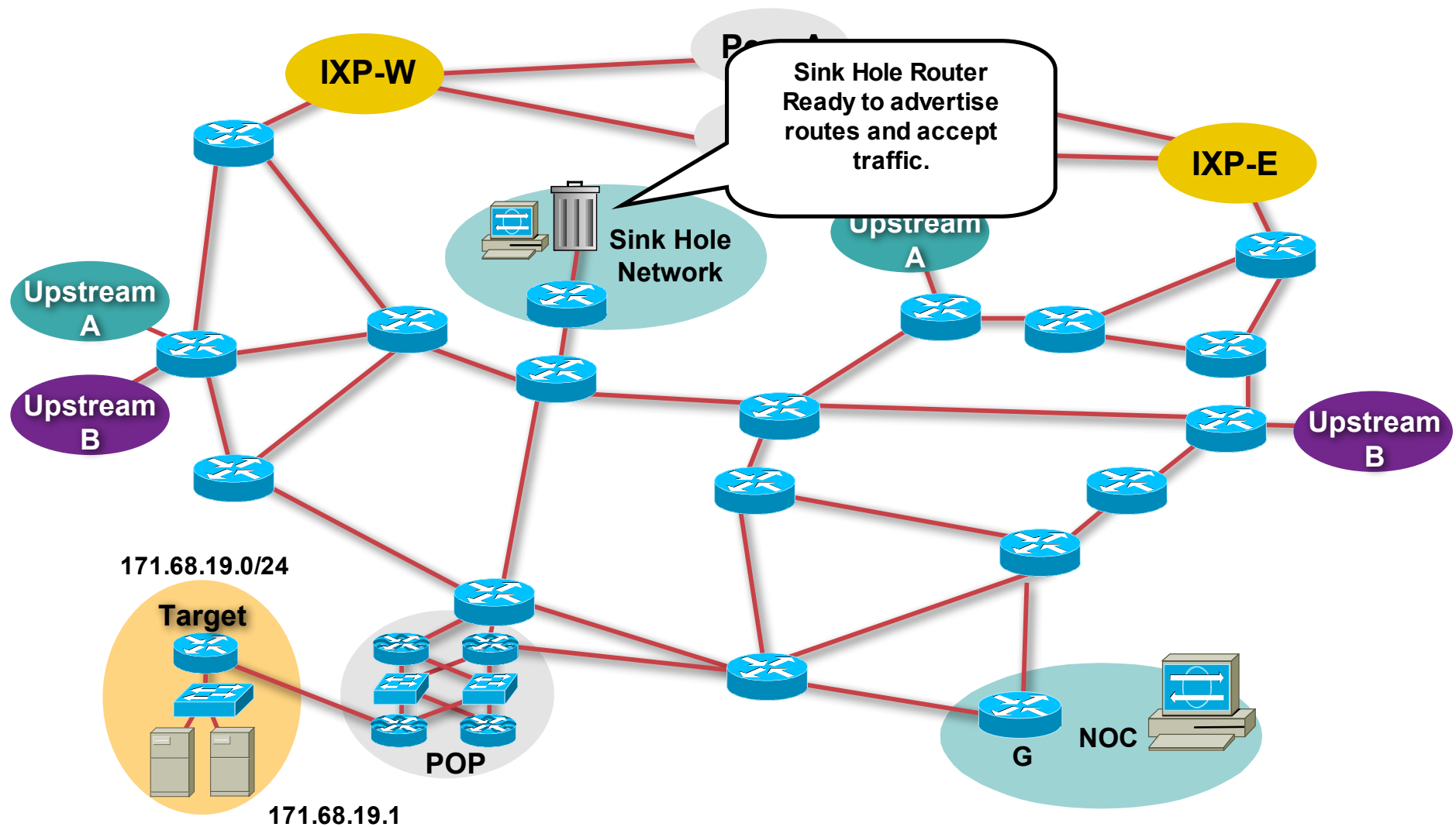
Code Red/NIMDA

Backscatter

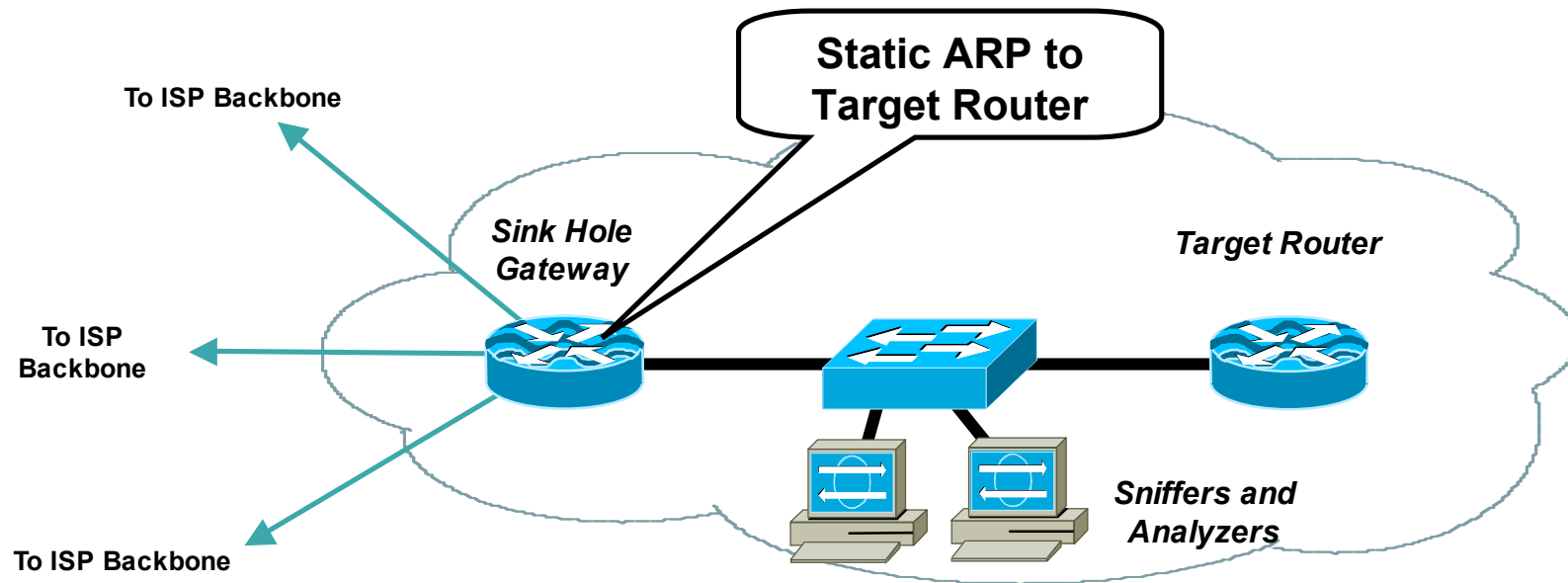
- Can place tracking tools and IDA in the Sink Hole network to monitor the noise.



Sink Hole Routers/Networks

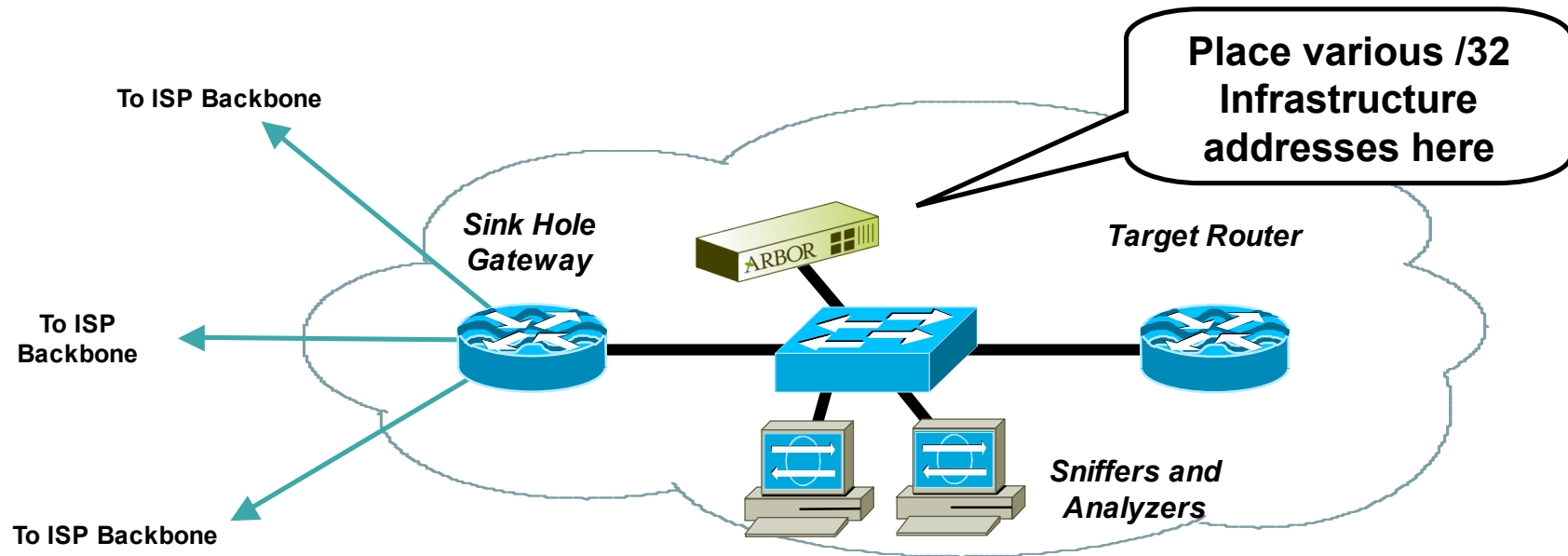


Target Routers are Expendable



- **Sink Hole Gateway Generates the more specific iBGP Announcement.**
- **Pull the DOS/DDOS attack to the sink hole and forwards the attack to the target router.**
- **Static ARP to the target router keeps the Sink Hole Operational**
– Target Router can crash from the attack and the static ARP will keep the gateway forwarding traffic to the ethernet switch.

Monitoring Scan Rates

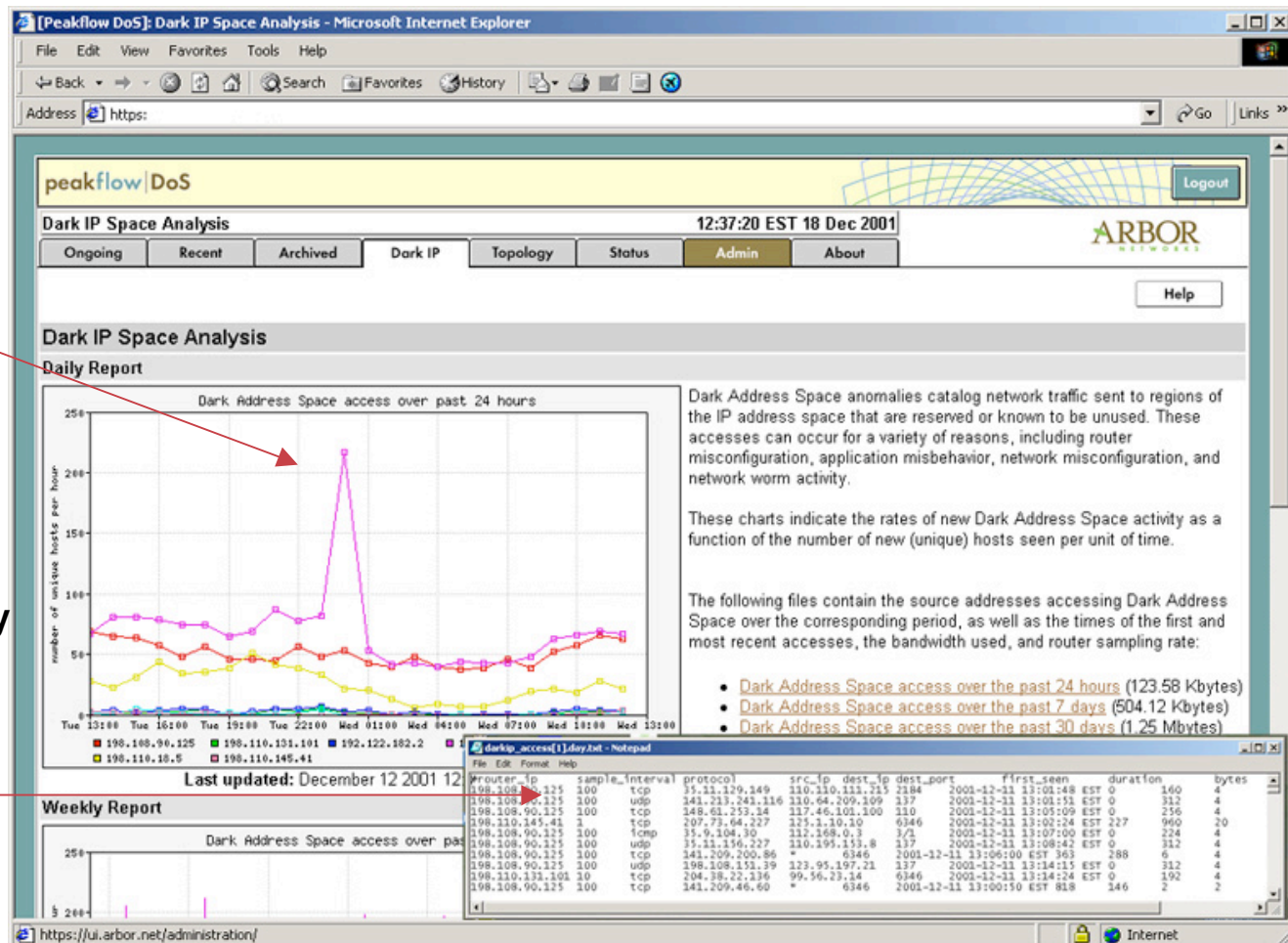


- **Select /32 address from different block of your address space. Advertise them out the Sink Hole**
- **Assign them to a workstation built to monitor and log scans.**
- **Arbor Network's *Dark IP* Application is one turn key commercial tool that can monitor scan rates.**

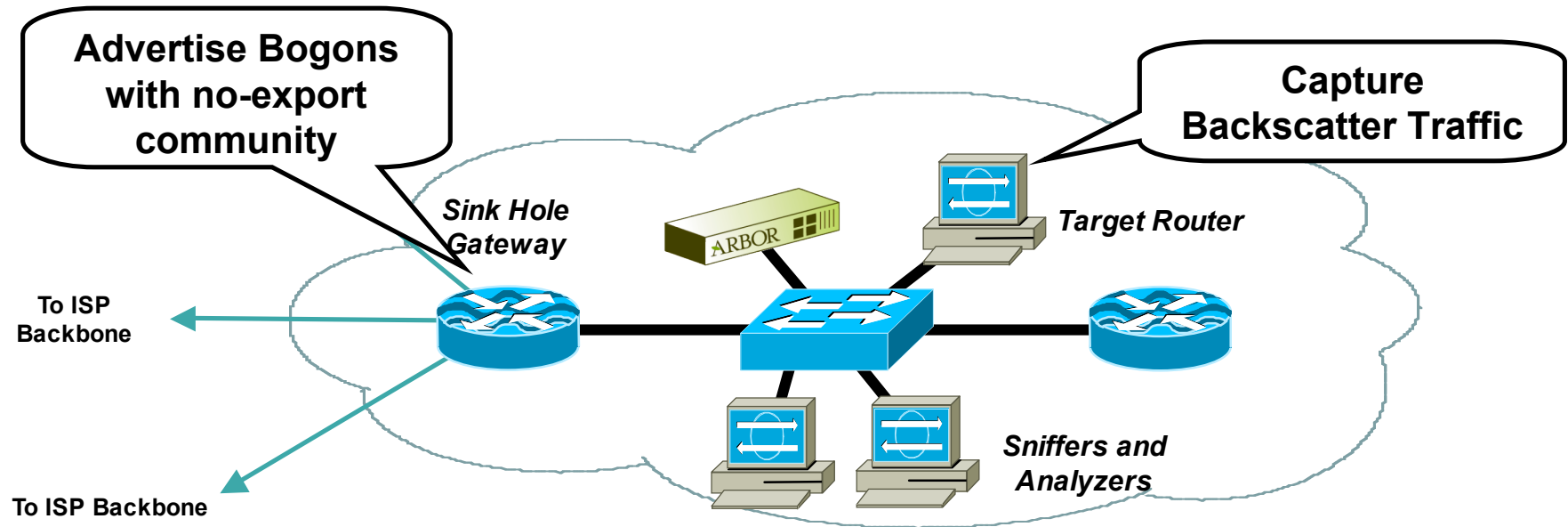
Worm Detection & Reporting UI

Operator instantly notified of Worm infection.

System automatically generates a list of infected hosts for quarantine and clean-up.



Monitoring Backscatter

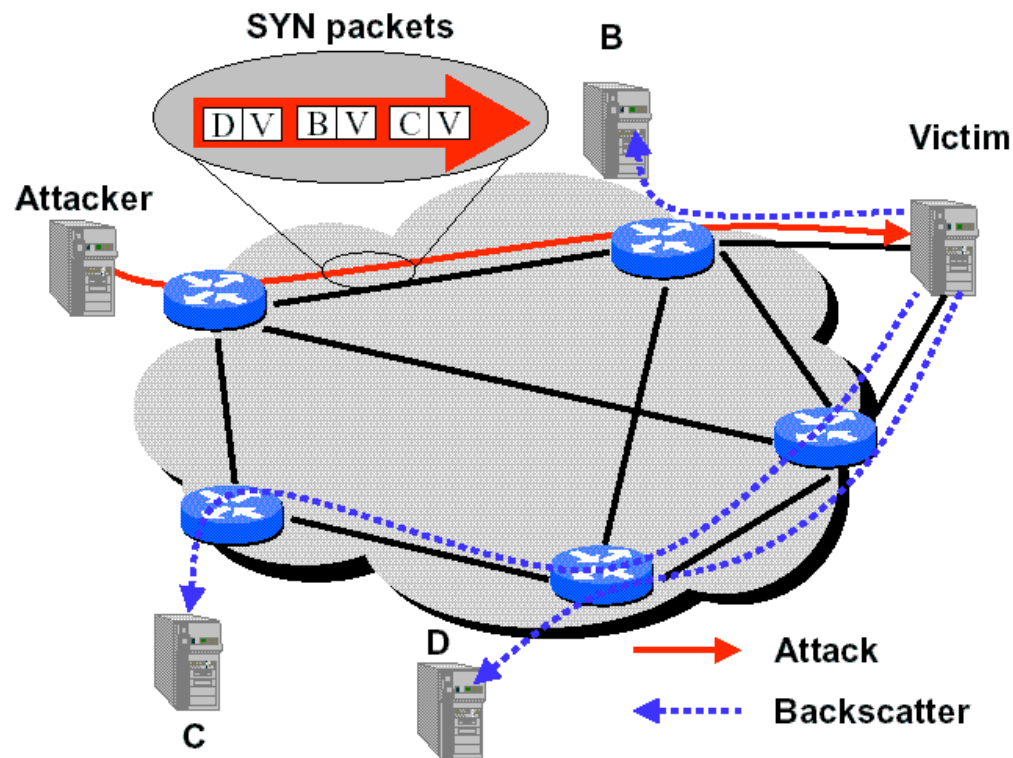


- Advertise bogon blocks with no-export and a safety community (plus ISP egress filtering on the edge)
- Static the bogon to a backscatter collector workstation (as simple as TCPdump).
- Pulls in backscatter for that range – allows monitoring.

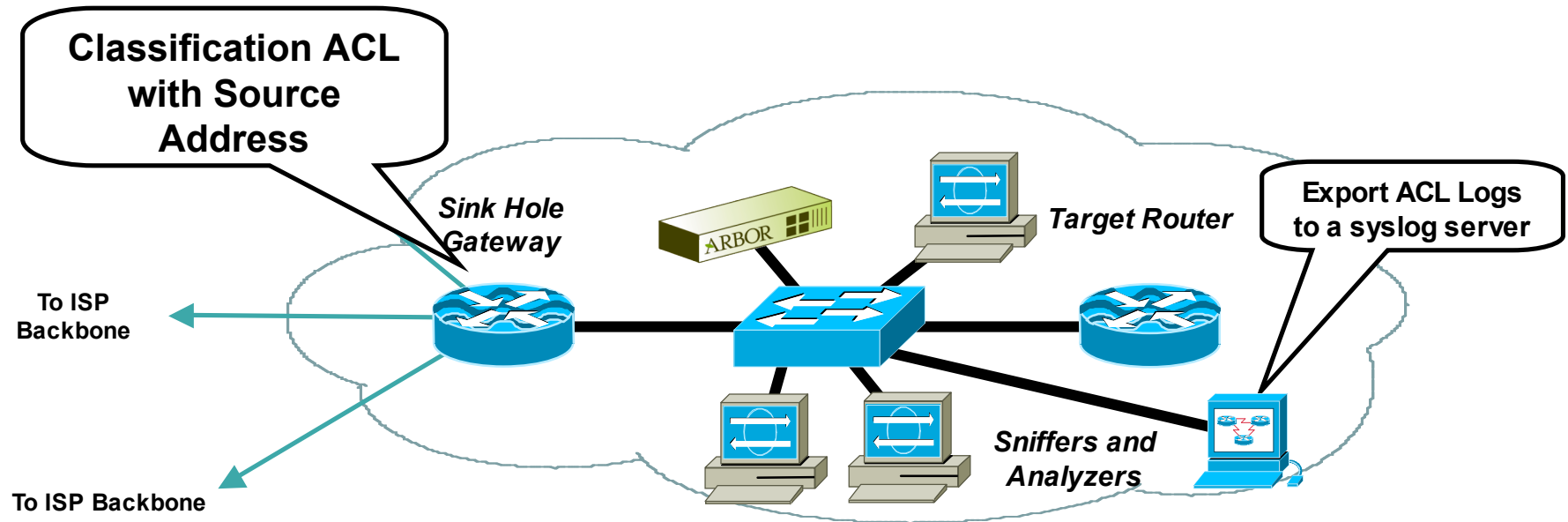
Monitoring Backscatter

- Inferring Internet Denial-of-Service Activity

<http://www.caida.org/outreach/papers/2001/BackScatter/>



Monitoring Spoof Ranges



- Hackers use ranges of valid (allocated blocks) and invalid (bogon, martin, and RFC1918 blocks) spoofed IP addresses.
- Extremely helpful to know the spoof ranges.
- Set up a classification filter on source addresses.

Monitoring Spoof Ranges

Example: Jeff Null's [jnull@truerouting.com] Test

```
Extended IP access list 120 (Compiled)

 permit tcp any any established (243252113 matches)

 deny ip 0.0.0.0 1.2.5.5.2.5.5.2.5.5 any (825328 matches)

 deny ip 2.0.0.0 0.2.5.5.2.5.5.2.5.5 any (413487 matches)

 deny ip 5.0.0.0 0.2.5.5.2.5.5.2.5.5 any (410496 matches)

 deny ip 7.0.0.0 0.2.5.5.2.5.5.2.5.5 any (413621 matches)

 deny ip 10.0.0.0 0.2.5.5.2.5.5.2.5.5 any (1524547 matches)

 deny ip 23.0.0.0 0.2.5.5.2.5.5.2.5.5 any (411623 matches)

 deny ip 27.0.0.0 0.2.5.5.2.5.5.2.5.5 any (414992 matches)

 deny ip 31.0.0.0 0.2.5.5.2.5.5.2.5.5 any (409379 matches)

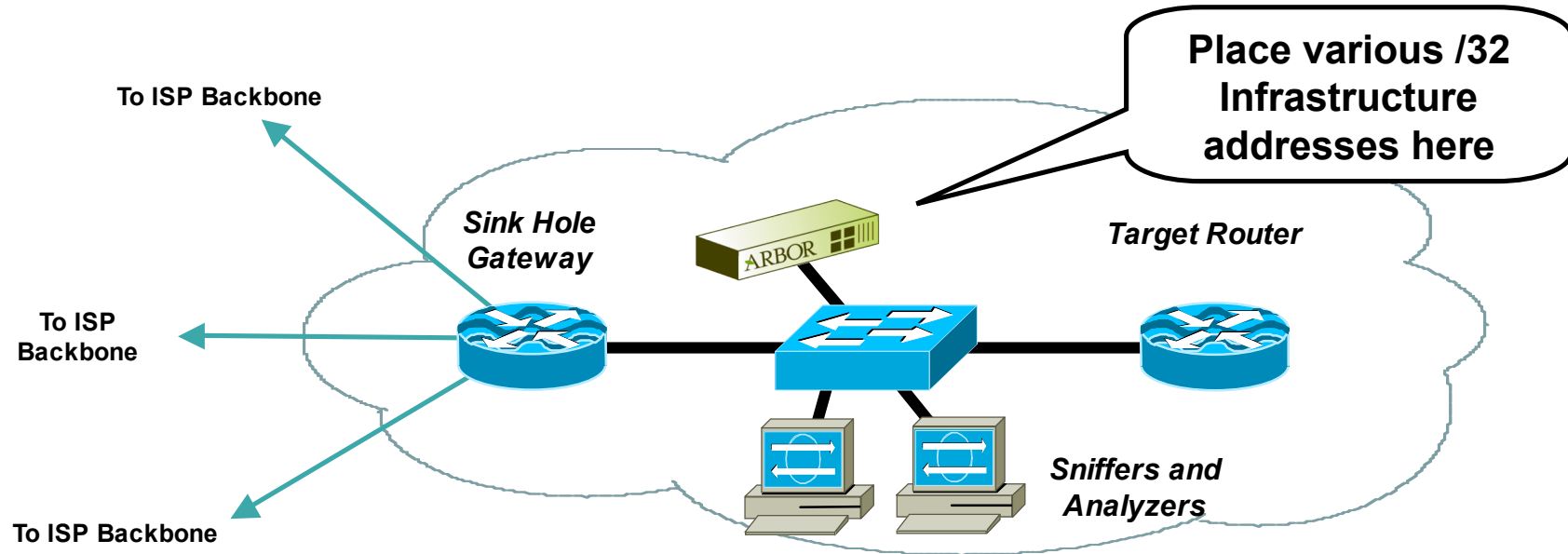
 deny ip 36.0.0.0 1.2.5.5.2.5.5.2.5.5 any (822904 matches)

.

.

 permit ip any any (600152250 matches)
```

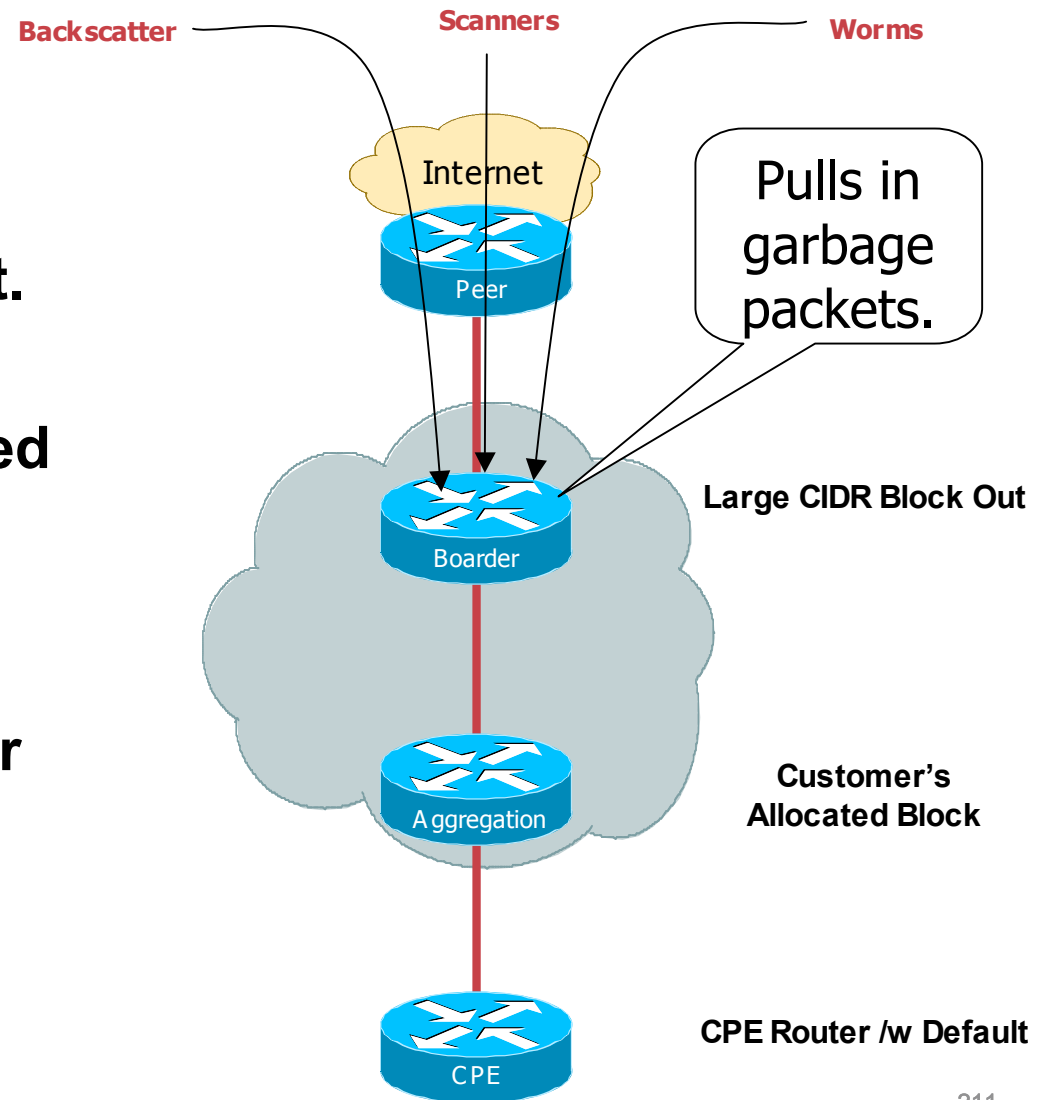
Monitoring Spoof Ranges



- **Select /32 address from different block of your address space. Advertise them out the Sink Hole**
- **Assign them to a workstation built to monitor and log scans.**
- **Arbor Network's *Dark IP* Application is one turn key commercial tool that can monitor scan rates.**

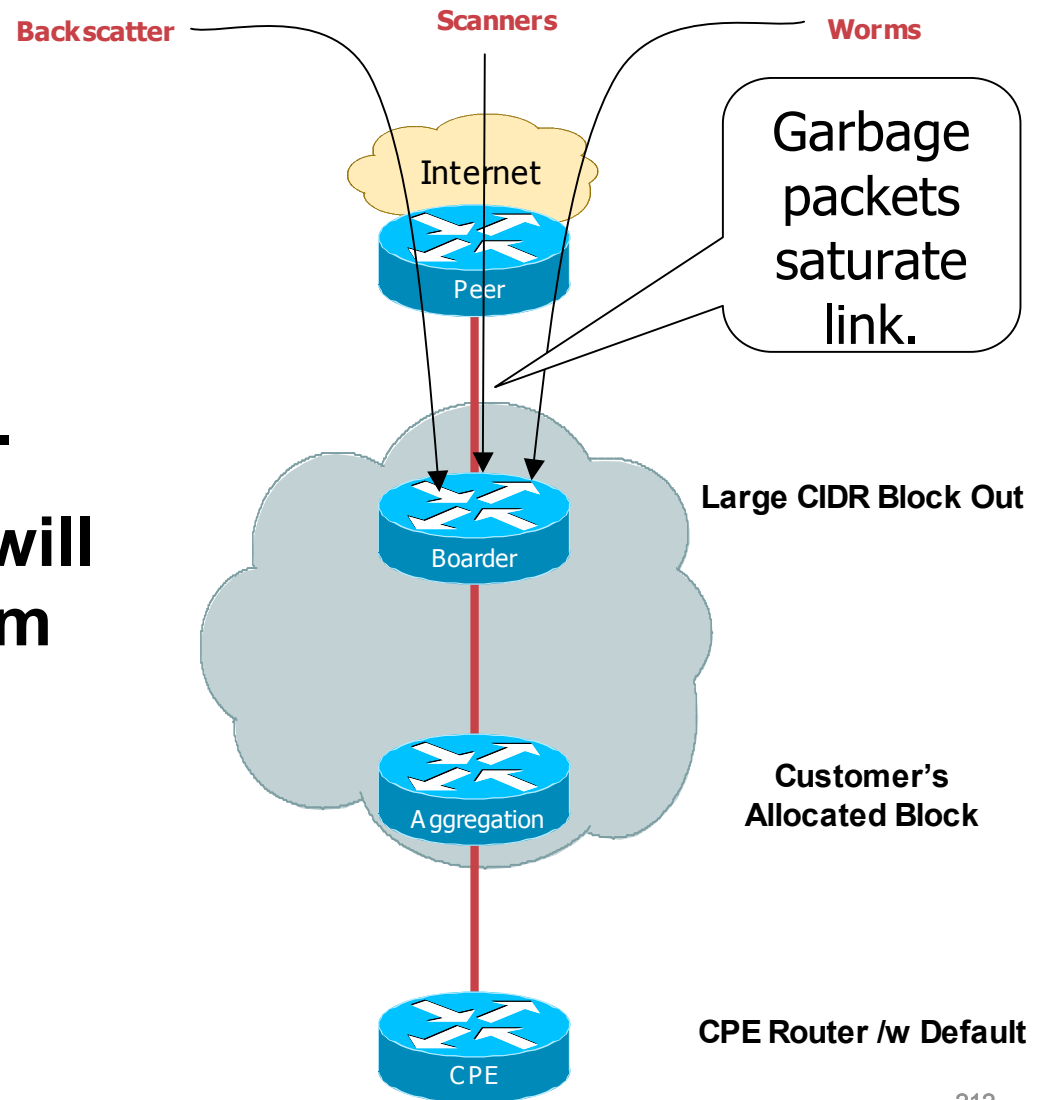
Simple Sink Holes – Internet Facing

- BCP is to advertise the whole allocated CIDR block out to the Internet.
- Left over unallocated Dark IP space gets pulled into the advertising router.
- The advertising router becomes a Sink Hole for garbage packets.



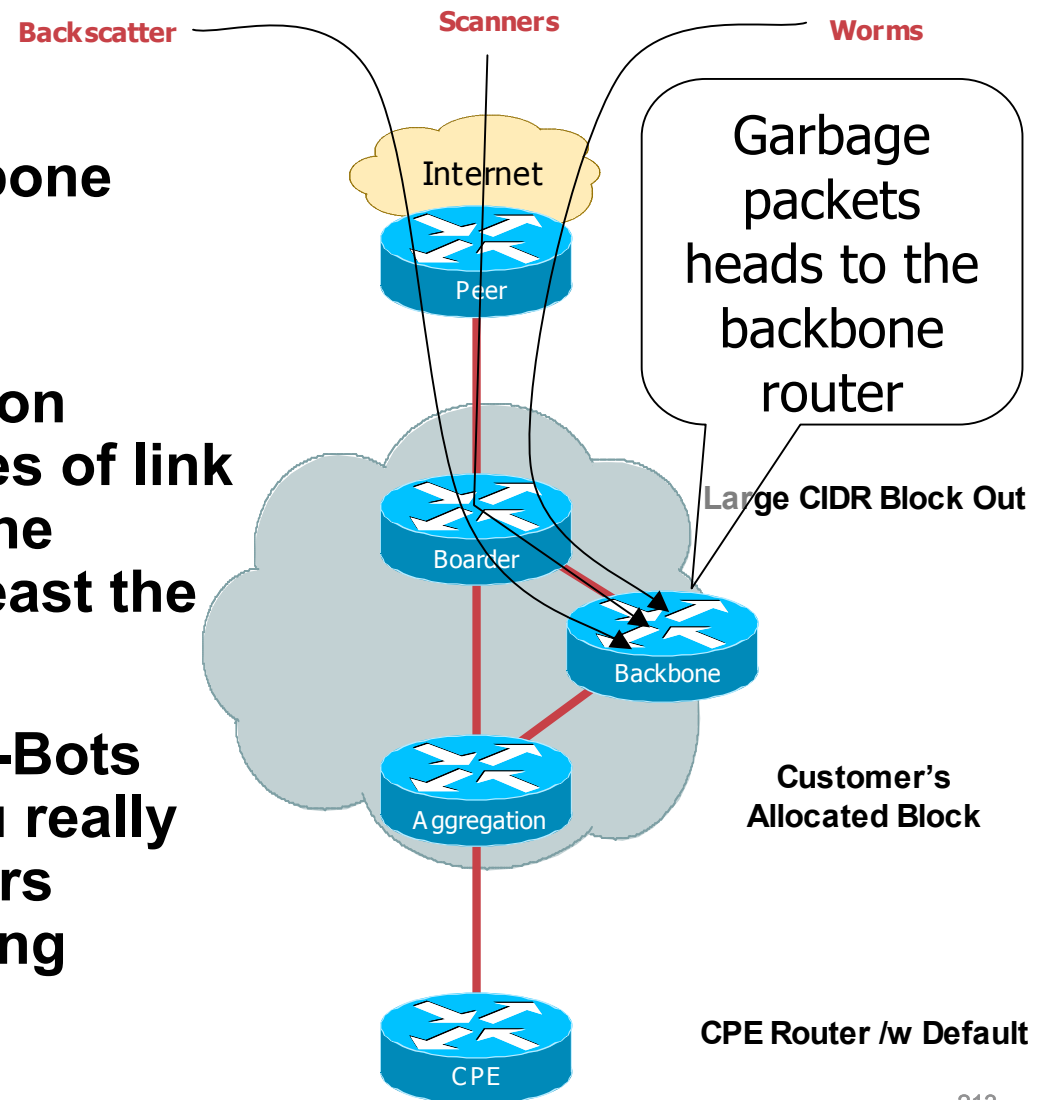
ASIC Drops at Line Rate?

- Forwarding/Feature ASICs will drop packets with no performance impact.
- Line Rate dropping will not solve the problem of garbage packets saturating the link.



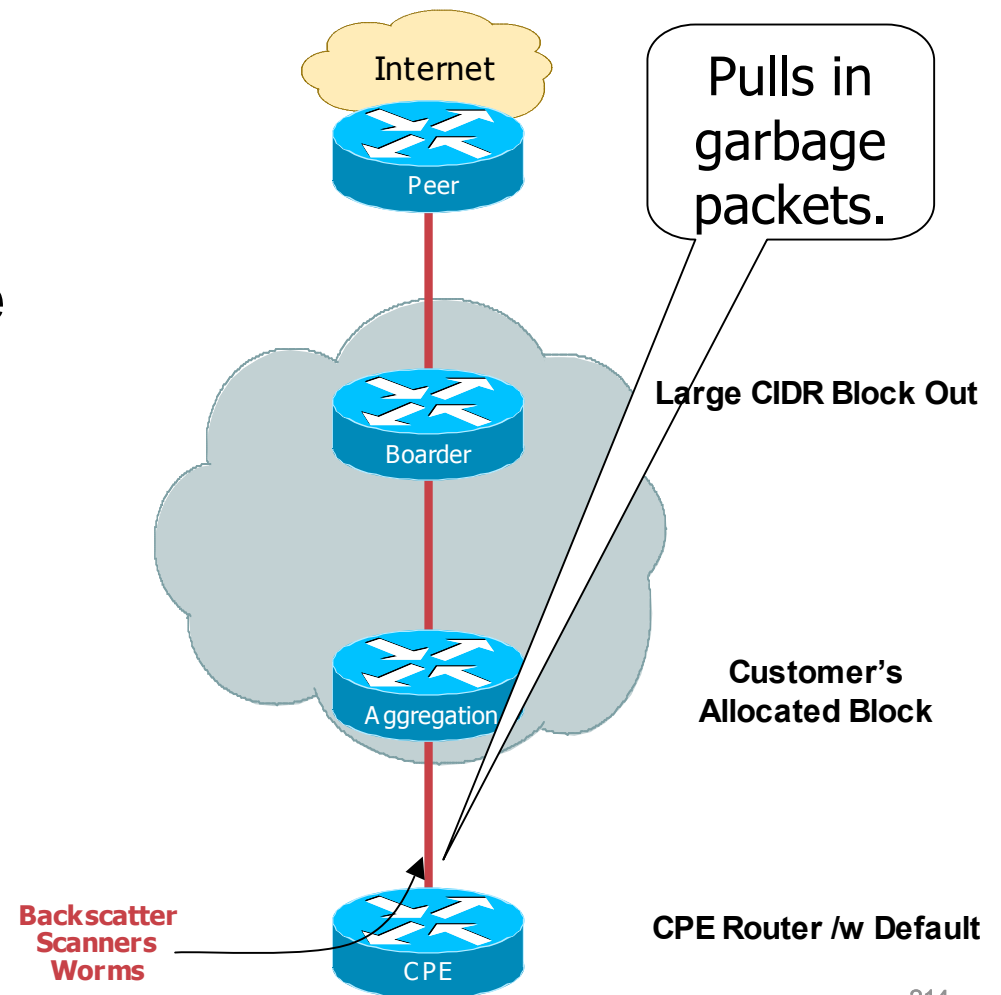
Backbone Router Injecting Aggregates

- Some ISPs use the Backbone Routers to Inject their Aggregates.
- Multiple Backbone injection points mitigates the issues of link saturation, but exposes the loopback addresses (at least the way it is done today).
- In a world of multiple Gig-Bots and Turbo worms, do you really want you backbone routers playing the role of dropping garbage packets?



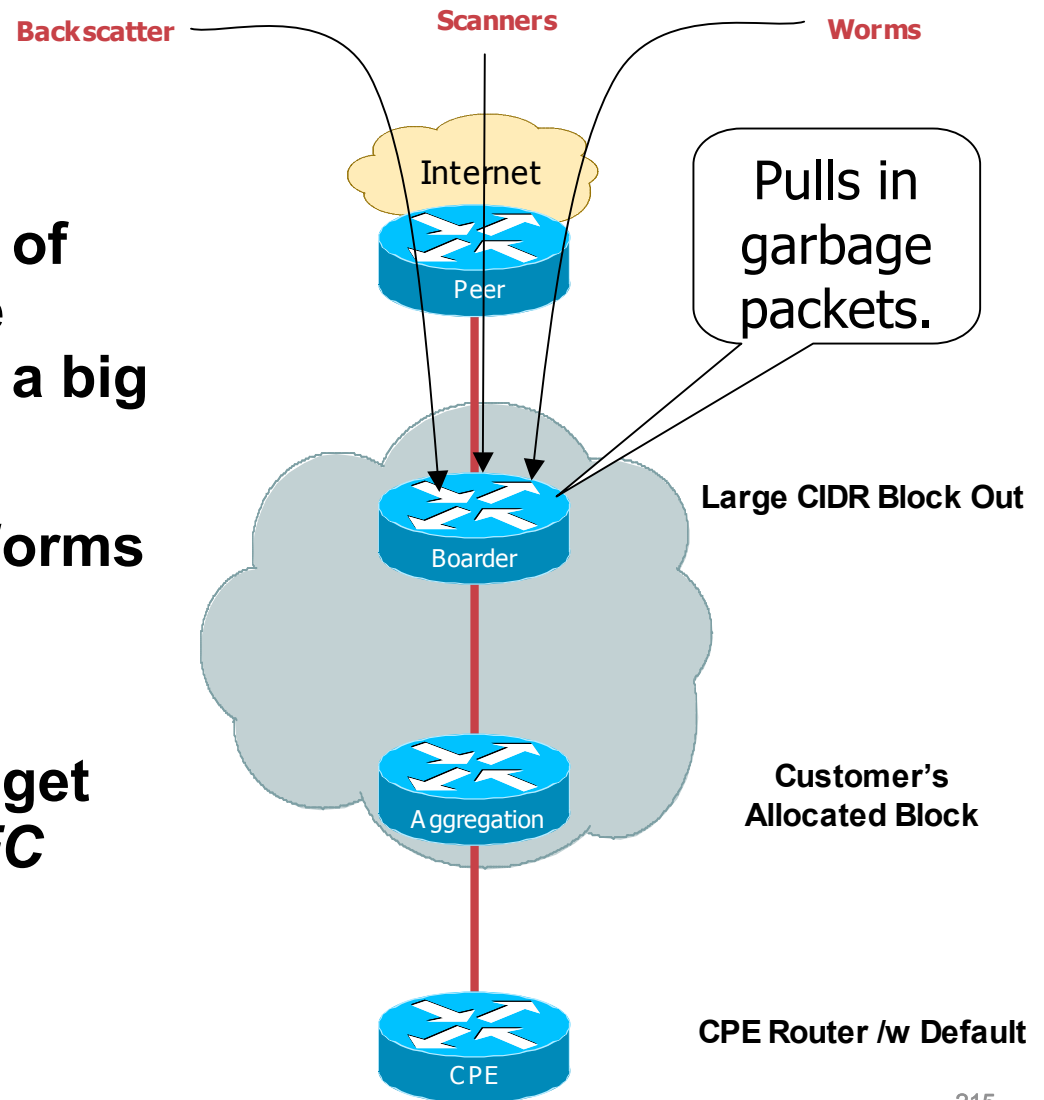
Simple Sink Holes – Customer Facing

- Defaults on CPE devices pull in everything.
- Default is the ultimate packet vacuum cleaner
- Danger to links during times of security duress.

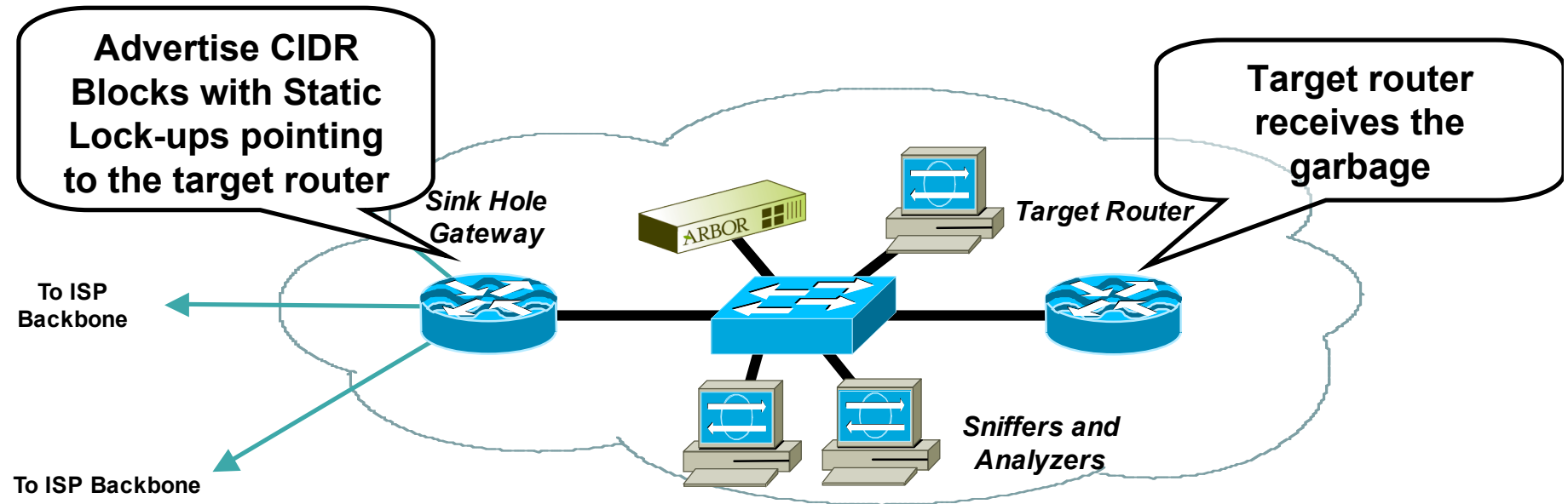


Simple Sink Holes – Impact Today

- In the past, this issue of pulling down garbage packets has not been a big deal.
- GigBots and Turbo Worms changes everything
- Even ASIC based forwarding platforms get impacted from the *RFC 1812 overhead*.

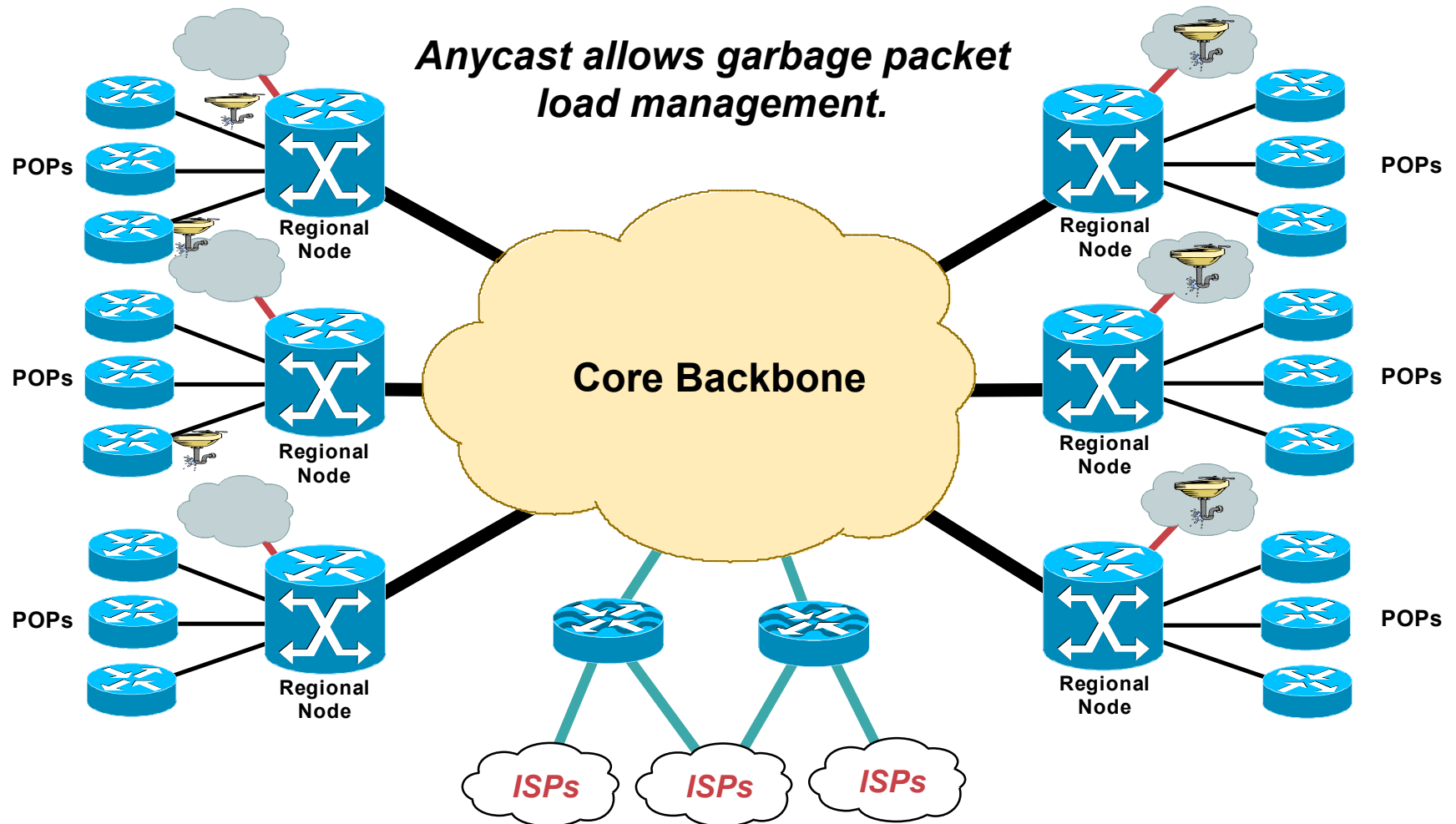


Sink Holes – Advertising Dark IP



- Move the CIDR Block Advertisements to Sink Holes.
- Does not impact BGP routing – route origination can happen anywhere in the iBGP mesh.
- Manages where you drop the packet.
- Turns the packet into a security tool.

Anycast Sink Holes to Scale



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 (224-239)**
 - RFC3330 defines special use IPv4 addressing**

Filtering Fragments

- Fragments can be explicitly denied
- Fragment handling is enabled via fragments keyword
- Default permit behavior → permit fragments that match ACE L3 entries
- Denies fragments and classifies fragment by protocol:

```
a c c e s s - l i s t 1 1 0 d e n y t c p a n y a n y f r a g m e n t s
```

```
a c c e s s - l i s t 1 1 0 d e n y u d p a n y a n y f r a g m e n t s
```

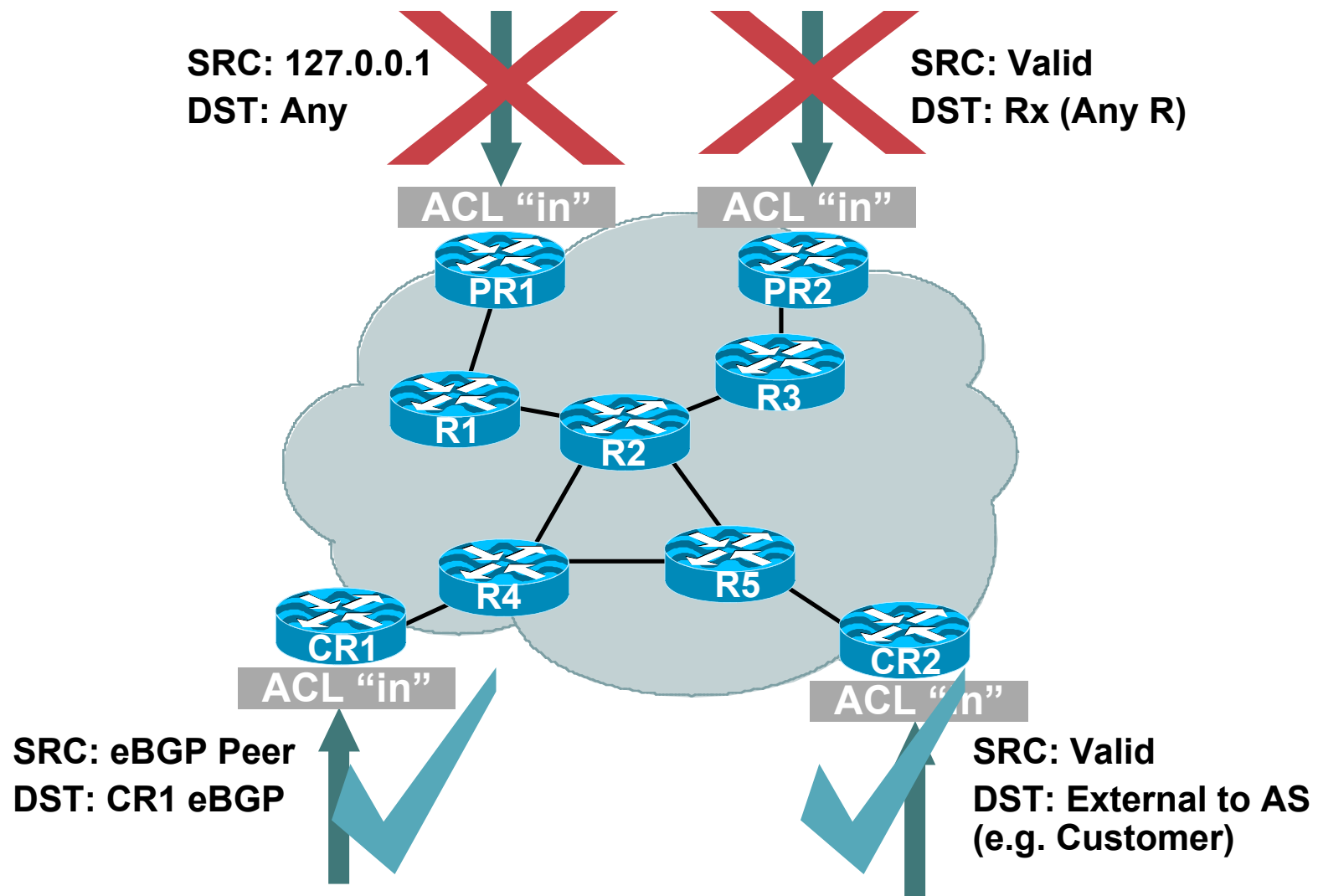
```
a c c e s s - l i s t 1 1 0 d e n y i c m p a n y a n y f r a g m e n t s
```

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**
- **Note: `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>
```

Infrastructure ACL in Action



Iterative Deployment: Step 1

- **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**

Step 1: IP Protocols

- **TCP—BGP, SSH, SSL**
- **UDP—SNMP, NTP, DNS**
- **IGP—OSPF, EIGRP**
- **GRE, IPv6 Tunneling**
- **ICMP to/from core routers**

ICMP unreachable/TTL expired for traceroute

Do you require other ICMP? (e.g. echo and echo-reply)

Caution: ICMP can be used for DoS

Step 1: IP Protocols

- **IPSec (ESP and maybe AH) + IKE**
- **Others?**
- **How many of these come from outside and terminate on your infrastructure?**

Step 1: Classification ACL

- **Classification ACL is 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**

Unexpected results should be carefully analyzed → do not permit protocols that you can't explain

Step 1: Classification ACL

- **Example:**

```
permit tcp any core_CIDR_block
permit udp any core_CIDR_block
permit gre any core_CIDR_block
permit esp any core_CIDR_block
permit ip any any
```

- **Classification ACLs affect data plane traffic**

All ACLs have implicit deny

Classification ACL must have permit any any to allow normal traffic to flow

Step 1: Classification ACL

- Use show access-list command to view ACE hit counts
- Example:

```
permit tcp any 10.86.183.0 0.0.0.255 (8 matches)
```
- Protocols that display hits should be reviewed to ensure that they are indeed required
 - Unexpected results should be analyzed
 - Needed protocols must be explicitly permitted
- Log keyword can be used as well to help provide details

Step 2: Begin to Filter

- Permit protocols identified in Step 1 to infrastructure only address blocks
- Deny all other to addresses blocks

Watch 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

- **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**
- **Increase security: deploy destination address filters if possible**

Example: Infrastructure ACL

! Deny our internal space as a source of external packets

```
a c c e s s - l i s t 1 0 1 d e n y i p o u r _ C I D R _ b l o c k a n y
```

! Deny src addresses of 0.0.0.0 and 127/8

```
a c c e s s - l i s t 1 0 1 d e n y i p h o s t 0 . 0 . 0 . 0 a n y
```

```
a c c e s s - l i s t 1 0 1 d e n y i p 1 2 7 . 0 . 0 . 0 . 2 5 5 . 2 5 5 . 2 5 5 a n y
```

! Deny RFC1918 space from entering AS

```
a c c e s s - l i s t 1 0 1 d e n y i p 1 0 . 0 . 0 . 0 . 2 5 5 . 2 5 5 . 2 5 5 a n y
```

```
a c c e s s - l i s t 1 0 1 d e n y i p 1 7 2 . 1 6 . 0 . 0 . 1 5 . 2 5 5 . 2 5 5 a n y
```

```
a c c e s s - l i s t 1 0 1 d e n y i p 1 9 2 . 1 6 8 . 0 . 0 . 0 . 2 5 5 . 2 5 5 a n y
```

Example: Infrastructure ACL

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

```
a c c e s s - l i s t 1 0 1 p e r m i t t c p h o s t p e e r A h o s t p e e r B e q 1 7 9
```

```
a c c e s s - l i s t 1 0 1 p e r m i t t c p h o s t p e e r A e q 1 7 9 h o s t p e e r B
```

! Deny all other access to infrastructure

```
a c c e s s - l i s t 1 0 1 d e n y i p a n y c o r e _ C I D R _ b l o c k
```

! Permit all data plane traffic

```
a c c e s s - l i s t 1 0 1 p e r m i t i p a n y a n y
```

Infrastructure ACL: Blocking IP Options

- Provide control functions that may be required in some situations but unnecessary for most common IP communications
- Include provisions for time stamps, security, and special routing
- The option field is variable in length. There may be zero or more options
- Complete list and description of IP Options in RFC 791
- Options can be set when using extended ping

Router#ping

Protocol [ip]: ip

Target IP address: 10.1.1.1

...

Extended commands [n]: y

...

Loose, Strict, Record, Timestamp, Verbose[none]:

Infrastructure ACL: Blocking IP Options

- **ip access-list extended drop-ip-option—slower than drop/ignore**

deny ip any any option any-options
permit ip any any

or

- **ip options drop—preferred over access-list**

Consider deploying at enterprise edge

Available in 12.0(23)S, 12.3(19), 12.3(4)T and 12.2(25)S

or

- **ip options ignore—router ignores options**

ISP best practice when router doesn't need to process options

“ignore” not available on all routing platforms

Available in 12.0(23)S for GSR

- **ip options drop and ignore reference:**

http://www.cisco.com/en/US/products/sw/iosswrel/ps1829/products_feature_guide_09186a00801d4a94.html

Block All IP Options: Considerations

- **IP options not switched in hardware**
- **Require control plane software processing**
 - Process the options**
 - Rewrite IP header**
- **Drop and ignore reduce load on RP—switched in hardware**
- **Malformed IP options**
 - Processed before interface access-lists**
 - IP option access-lists don't apply**
 - Dropped by receiving router**
 - Router generates ICMP Type 12 messages (Self DoS)**

Block All IP Options: Considerations

- **Can be configured as part of CoPP default class**
CoPP protects against malformed IP options in 12.2(32.8)S, 12.4(7) and 12.4(6)T
- **Some legitimate protocols use options**
 - RSVP (NetMeeting)**
 - MPLS TE**
 - MPLS OAM**
 - IGMPv2**
 - IGMPv3**
 - DVMRP**
 - PGM**

ACL Support for Filtering on TTL Value

- Filter packets based on IP header time-to-live
- Interface TTL ACLs
 - TTL 0 or 1 denies are process switched
 - Use CoPP to mitigate TTL 0 or 1 drops
 - Other TTLs dropped in fastest switching path
- Denies prevent ICMP time-exceeded messages
- Example:

Extended IP access list ttl-drops

```
10 deny ip any any ttl range 0 1
```

```
20 permit ip any any
```

- Available in 12.4(2)T

<http://www.cisco.com/univercd/cc/td/doc/product/software/ios124/124newft/124t/124t2/htaclttl.htm>

Infrastructure ACL: TCP Flags Filtering

- ACL to block Nmap scanning

ip access-list extended block-nmap

remark block stealth fin scan

deny tcp any any match-all -ack +fin -psh -rst -syn -urg log

remark block xmas scan

deny tcp any any match-all +fin +psh +urg log

remark allow syn or ack which should block null scan

permit tcp any any match-any +ack +syn

deny tcp any any log

permit ip any any

- Available in 12.3(4)T and 12.2(25)S

http://www.cisco.com/en/US/products/ps6350/products_configuration_guide_chapter09186a0080431049.html

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

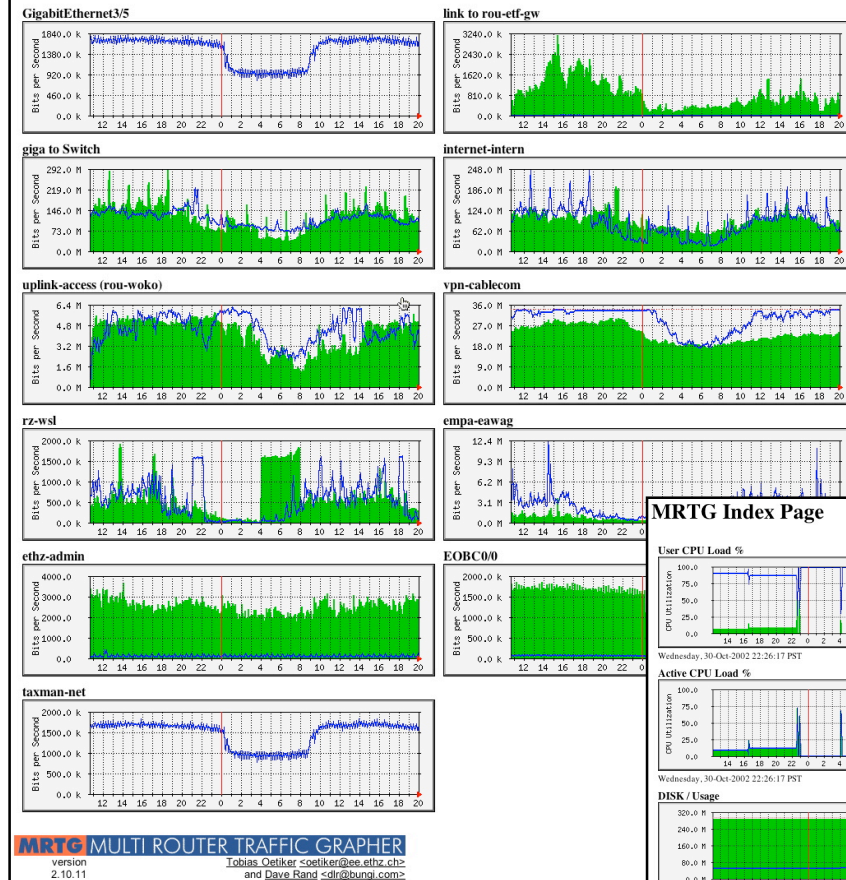
- Formerly known as UCD-SNMP toolset
- Open source SNMP command-line tools, library, trap-generator, agent, etc. available from <http://www.net-snmp.org/>
- Included with most Linux distros, FreeBSD, etc.
- Command-line access to SNMP data from enabled routers, switches, etc.
- Runs on Linux, FreeBSD, 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://people.ee.ethz.ch/~oetiker/webtools/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

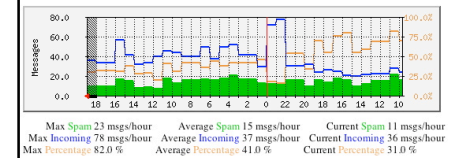
MRTG Index Page



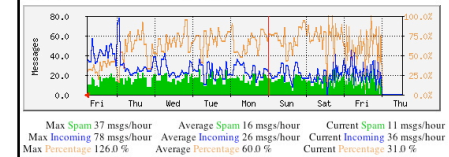
Incoming messages vs. spam per hour

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

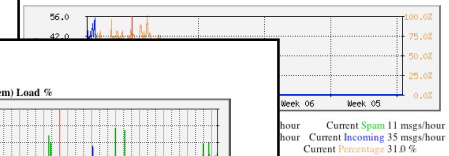
'Daily' Graph (60 Minute Average)



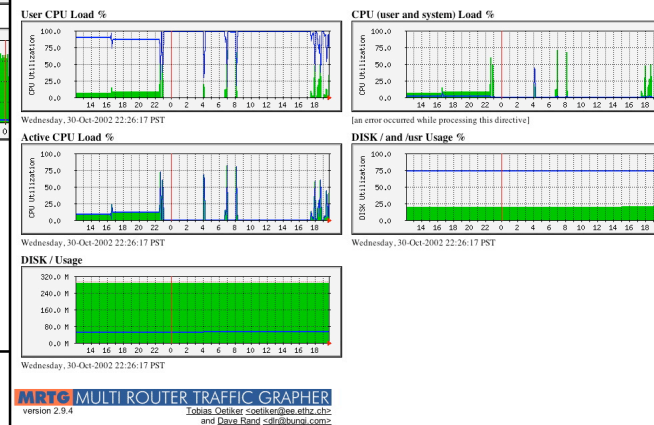
'Weekly' Graph (30 Minute Average)



'Monthly' Graph (2 Hour Average)



MRTG Index Page

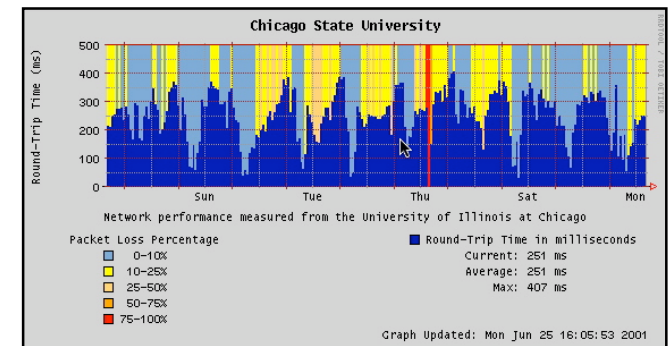
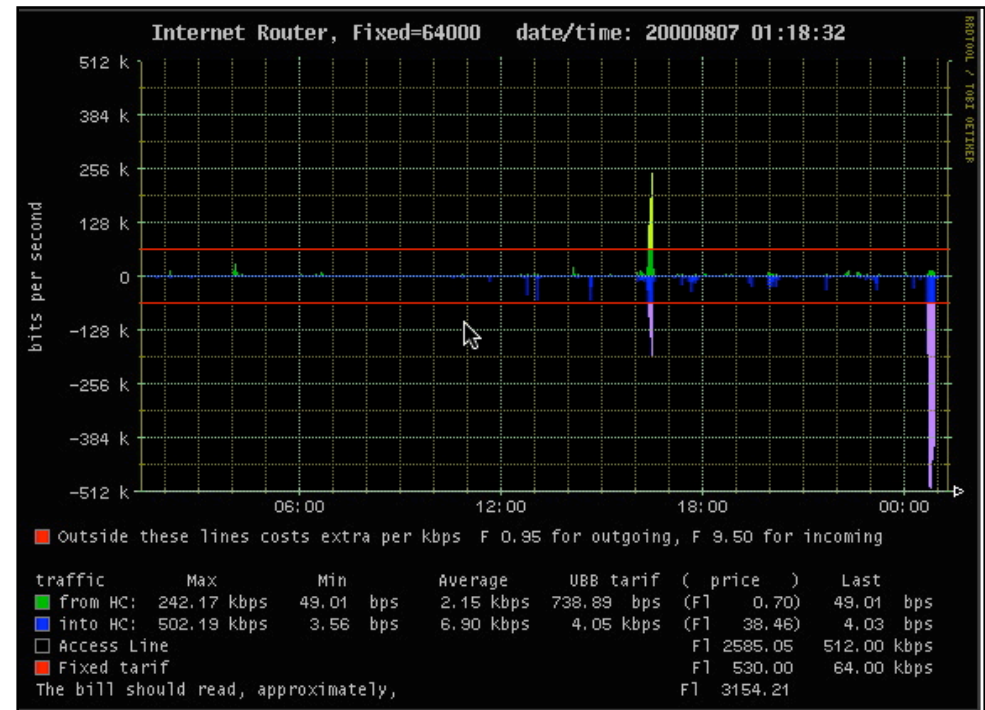
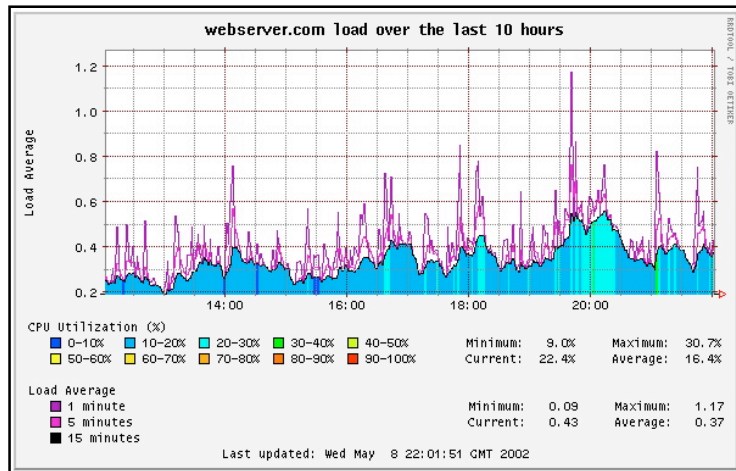
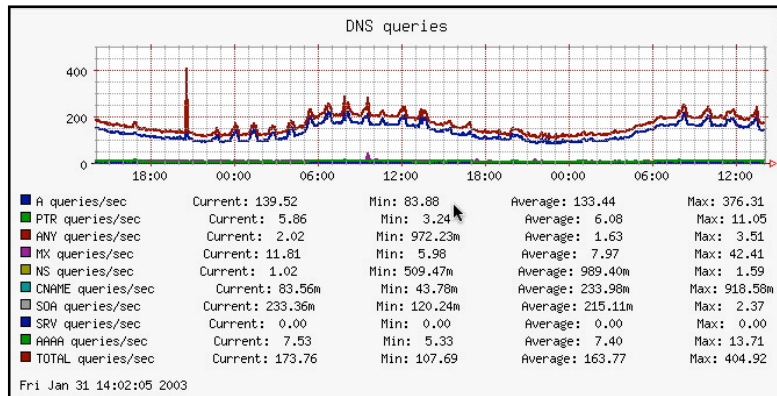


Source: mrtg.org

SNMP: RRDTool

- **RRDTool—the Round Robin Database Tool**
- **Another open source SNMP visualization toolset developed by Tobi Oetiker, available from**
<http://people.ee.ethz.ch/~oetiker/webtools/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



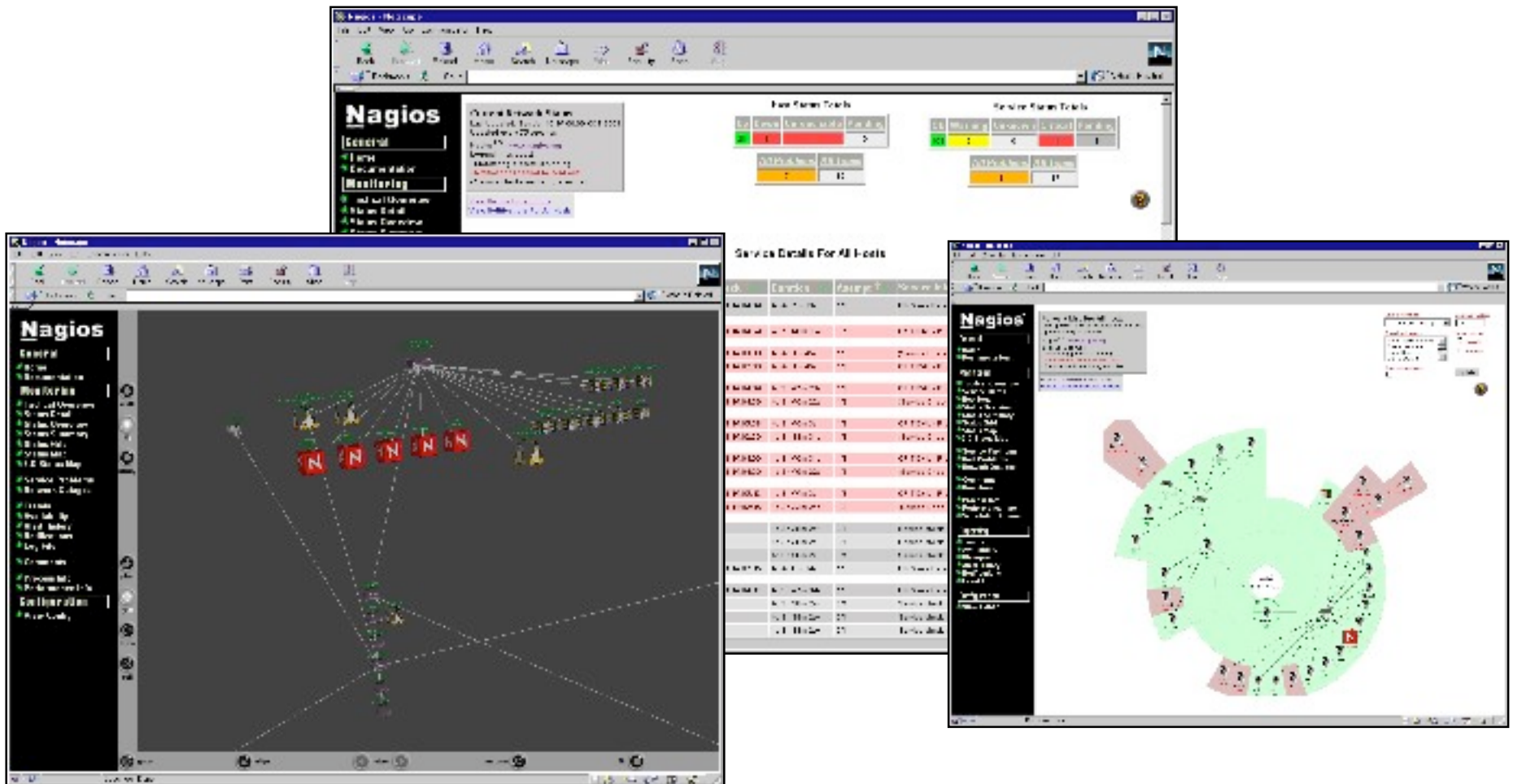
Source:

<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://bigsisiter.graeff.com/>), Nagios (<http://www.nagios.org/>), and others

Nagios Examples

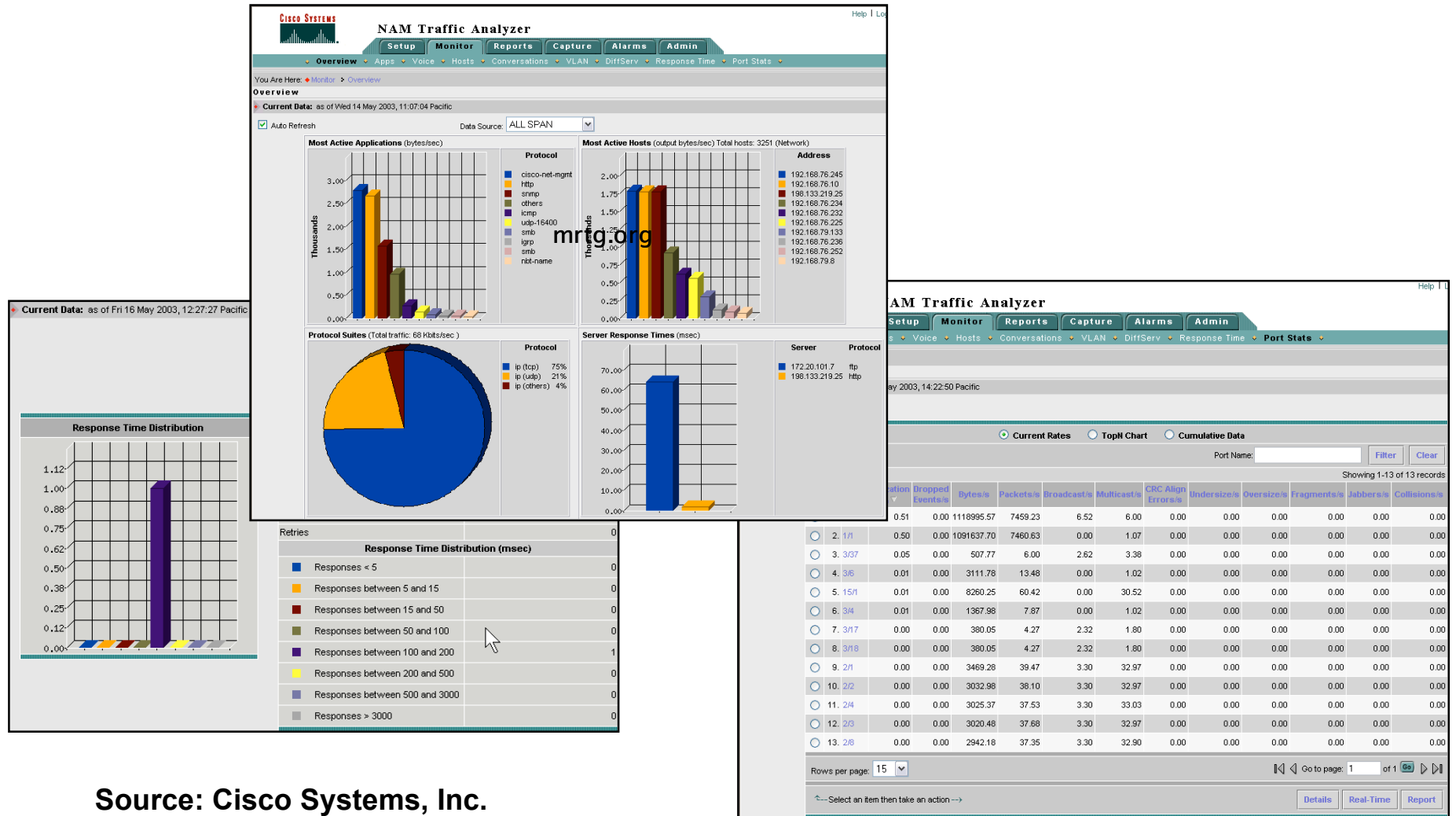


Source: <http://www.nagios.org>

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



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, Ethereal (<http://www.ethereal.com>) 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



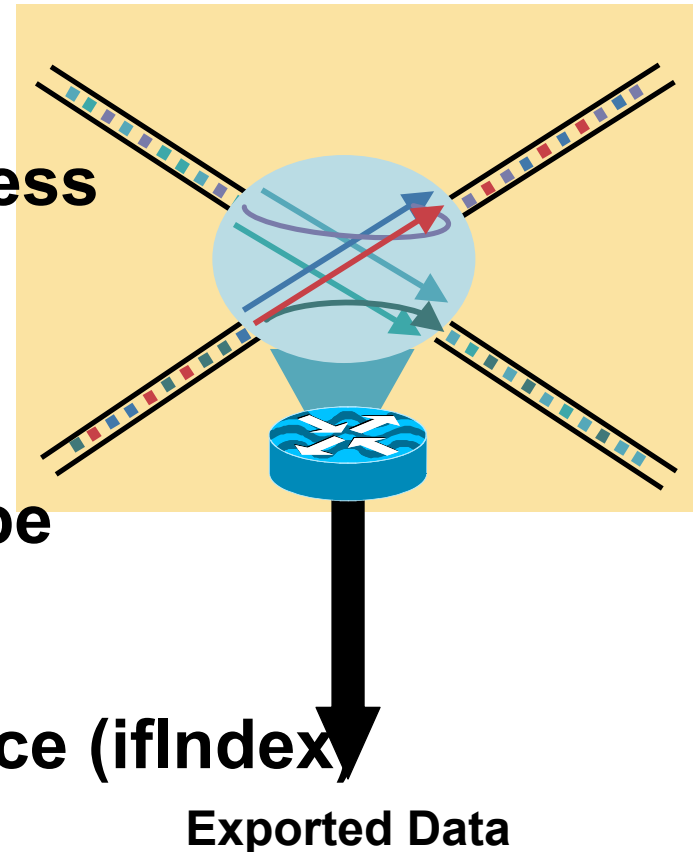
NetFlow Origination

- **Developed by Darren Kerr and Barry Bruins at Cisco Systems in 1996**
US Patent 6,243,667
- **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”**

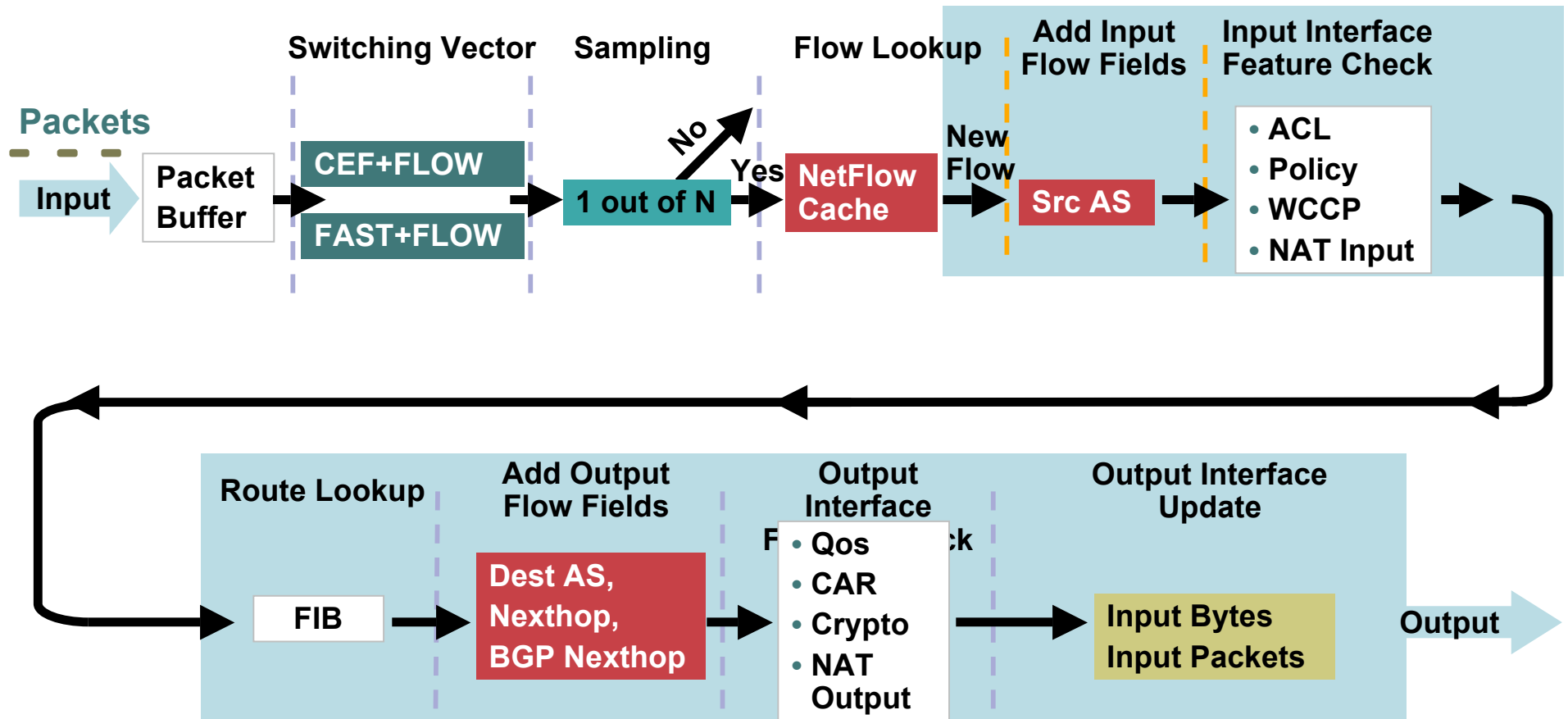
What Is a Flow?

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)



Switching Path



Cisco 1700 through 7300 Series Routers

NetFlow—Software-Based Platforms

1. Create and update flows in NetFlow Cache

SrcIf	SrcIPadd	DstIf	DstIPadd	Protocol	TOS	Flgs	Pkts	SrcPort	SrcMsk	SrcAS	DstPort	DstMsk	DstAS	NextHop	Bytes/Pkt	Active	Idle
Fa1/0	173.100.21.2	Fa0/0	10.0.227.12	11	80	10	11000	00A2	/24	5	00A2	/24	15	10.0.23.2	1528	1745	4
Fa1/0	173.100.3.2	Fa0/0	10.0.227.12	6	40	0	2491	15	/26	196	15	/24	15	10.0.23.2	740	41.5	1
Fa1/0	173.100.20.2	Fa0/0	10.0.227.12	11	80	10	10000	00A1	/24	180	00A1	/24	15	10.0.23.2	1428	1145.5	3
Fa1/0	173.100.6.2	Fa0/0	10.0.227.12	6	40	0	2210	19	/30	180	19	/24	15	10.0.23.2	1040	24.5	14

2. Expiration

- Inactive timer expired (15 seconds is default)
- Active timer expired (30 min (1,800 seconds) is default)
- NetFlow cache is full (oldest flows are expired)
- RST or FIN TCP Flag

SrcIf	SrcIPadd	DstIf	DstIPadd	Protocol	TOS	Flgs	Pkts	SrcPort	SrcMsk	SrcAS	DstPort	DstMsk	DstAS	NextHop	Bytes/Pkt	Active	Idle
Fa1/0	173.100.21.2	Fa0/0	10.0.227.12	11	80	10	11000	00A2	/24	5	00A2	/24	15	10.0.23.2	1528	1800	4

3. Aggregation?

No

Yes

e.g. Protocol-Port Aggregation Scheme becomes

Protocol	Pkts	SrcPort	DstPort	Bytes/Pkt
11	11000	00A2	00A2	1528

4. Export Version

Non-Aggregated Flows—Export Version 5 or 9

Aggregated Flows—Export Version 8 or 9

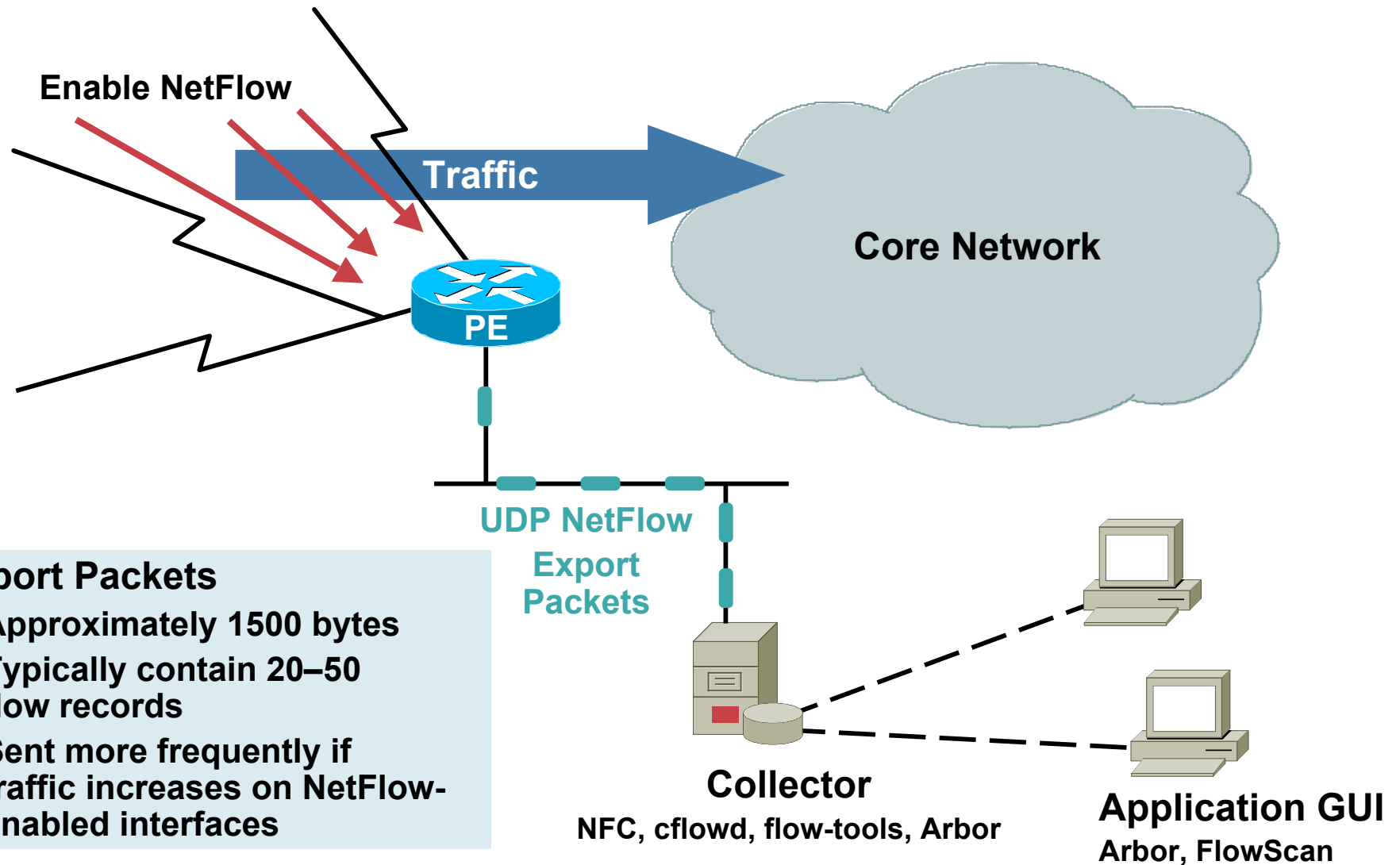
5. Transport Protocol

Export
Packet

Header

Payload
(Flows)

Creating Export Packets



Uses of NetFlow

Service Provider	Enterprise
<ul style="list-style-type: none">• Peering Arrangements• SLA VPN User Reporting• Usage-Based Billing• DoS/Worm Detection• Traffic Engineering• Troubleshooting	<ul style="list-style-type: none">• 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

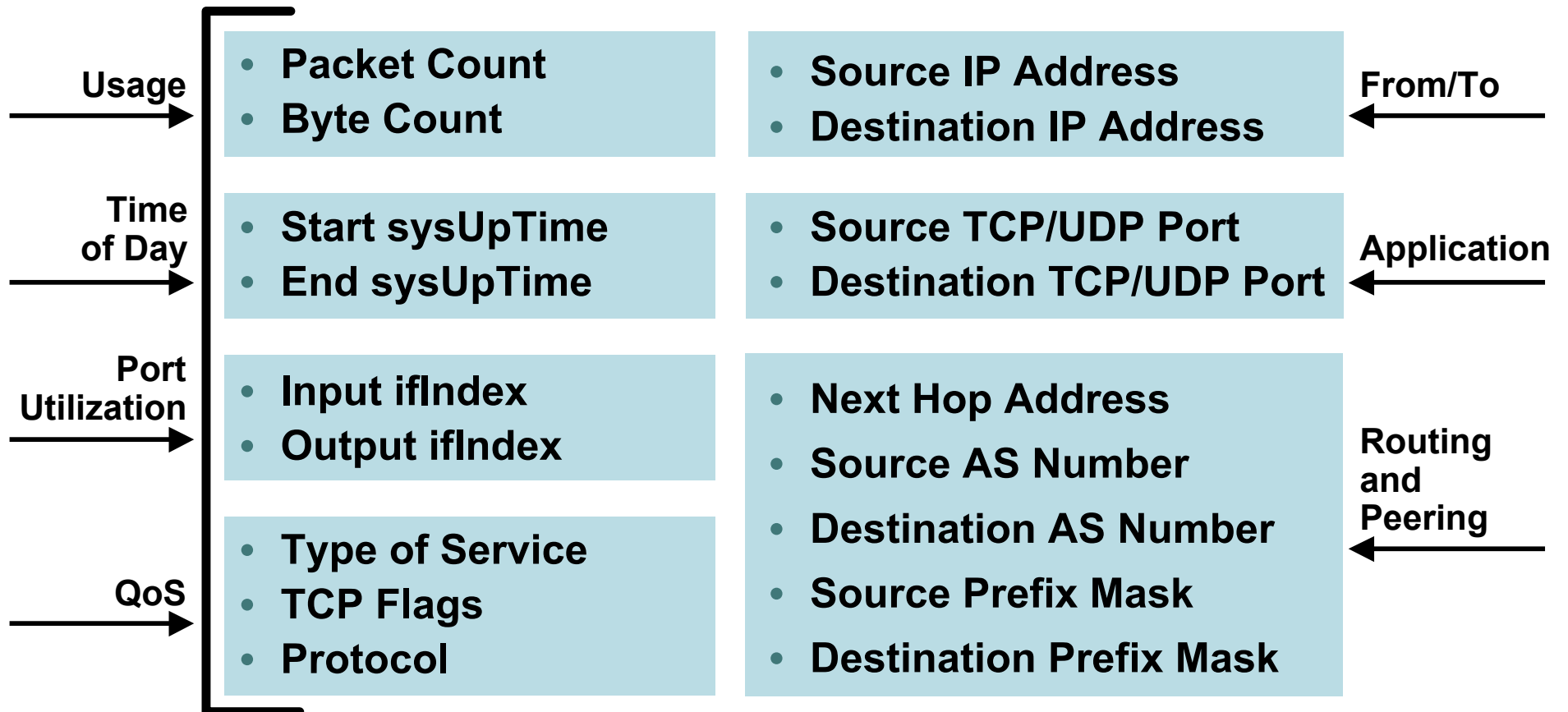
Key Concept: NetFlow Scalability

- 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

Version 5: Flow Format



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**

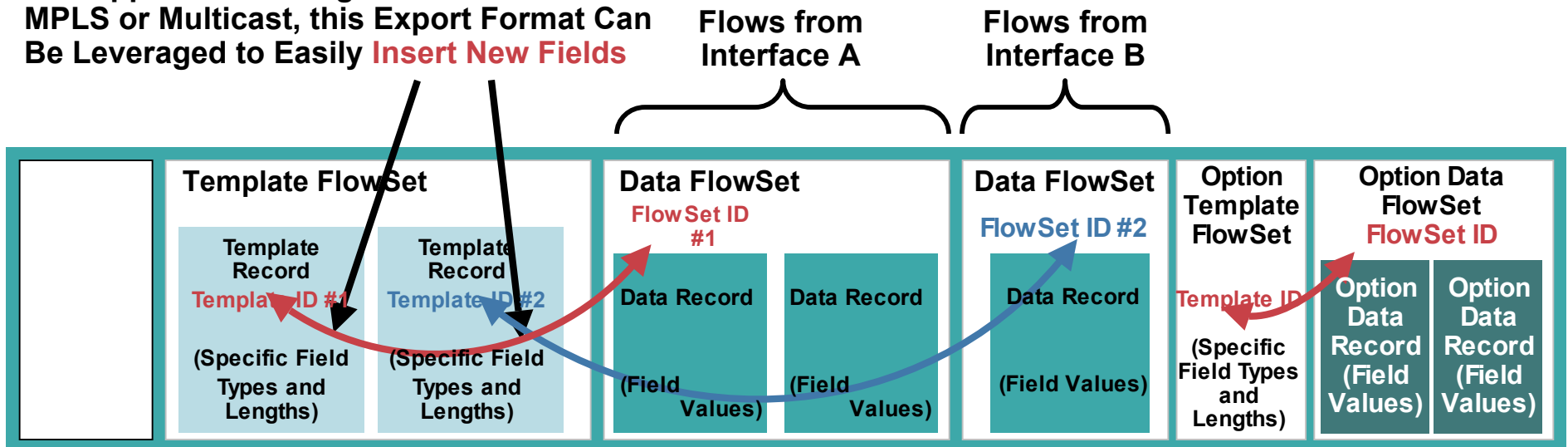
Solution: Build a **Flexible and **Extensible** Export Format**

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 v9 Export Packet

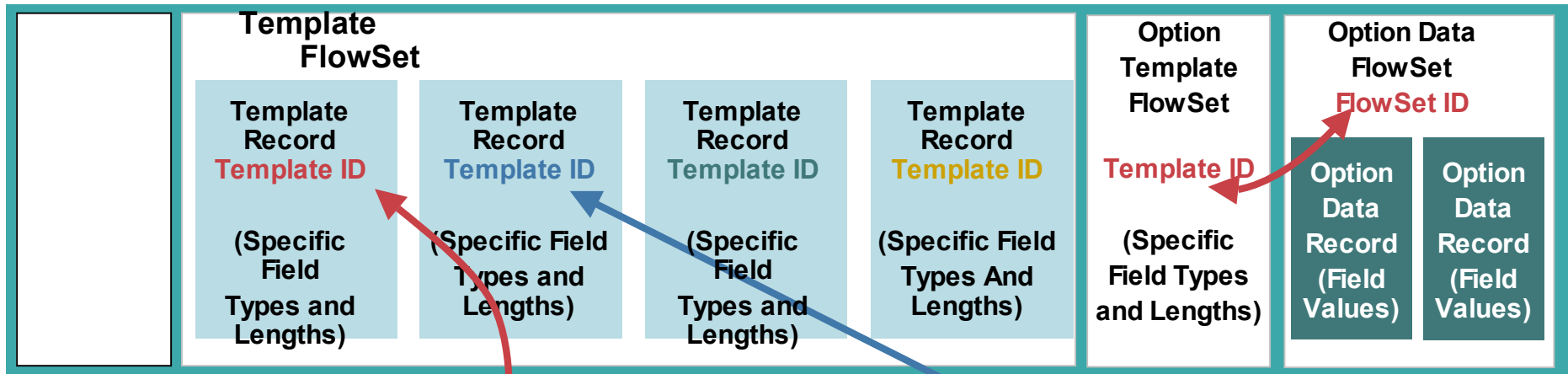
To Support Technologies such as MPLS or Multicast, this Export Format Can Be Leveraged to Easily **Insert New Fields**



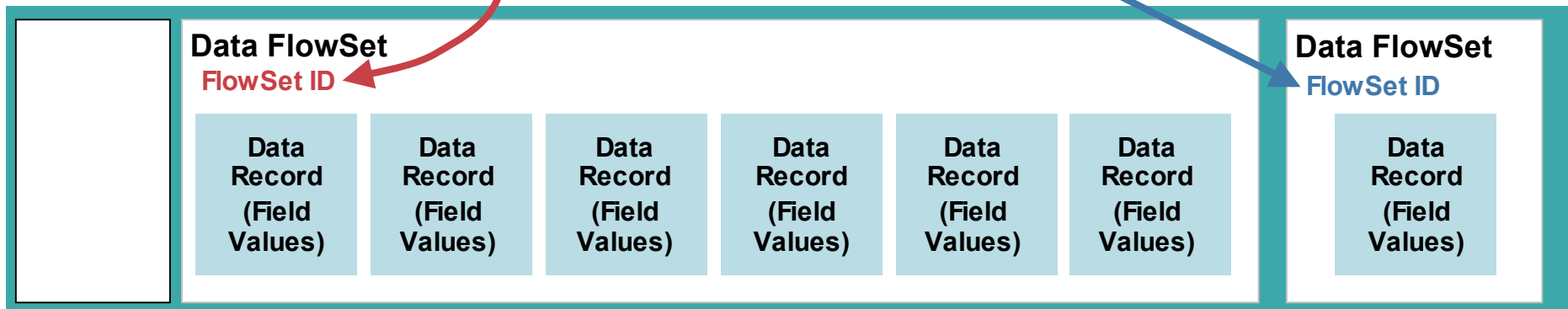
- Matching ID numbers is the way to associate template to the data records
- The header follows the same format as prior NetFlow versions so collectors will be backward compatible
- Each data record represents one flow
- If exported flows have the same fields then they can be contained in the same template record, e.g. unicast traffic can be combined with multicast records
- If exported flows have different fields then they can't be contained in the same template record, e.g. BGP next-hop can't be combined with MPLS aware NetFlow records

NetFlow v9 Flexible Format

Example of Export Packet Right After Router Boot or NetFlow Configuration



Example of Export Packets Containing Mostly Flow Information



NetFlow v9 Export

Configuring Version 9 Export

```
pamela(config)# ip flow-export version ?
```

1

5

9

**Export Versions Available for
Standard NetFlow Flows**

```
pamela(config)# ip flow-export version 9
```

Configuring Version 9 Export for an Aggregation Scheme

```
pamela(config)# ip flow-aggregation cache as
```

```
pamela(config-flow-cache)# enabled
```

```
pamela(config-flow-cache)# export ?
```

destination Specify the Destination IP address

version configure aggregation cache export version

```
pamela(config-flow-cache)# export version ?
```

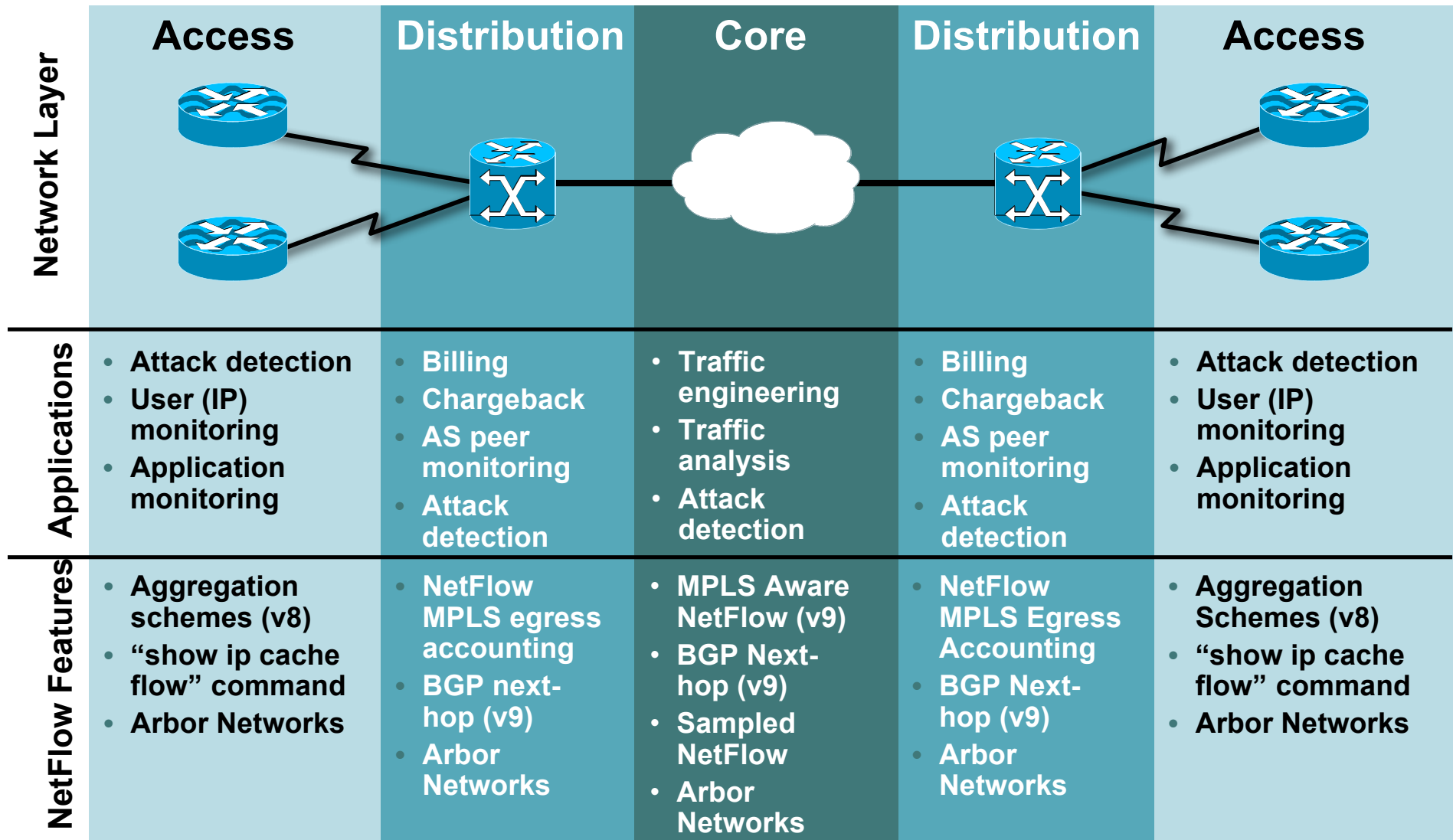
8 Version 8 export format

9 Version 9 export format

**Export Versions Available for
Aggregated NetFlow Flows**

```
pamela(config-flow-cache)# export version 9
```

NetFlow in the Topology



Cisco 7200 NetFlow Example

```

7200>sh ip cache flow

IP packet size distribution (14952M total packets):

  1-32    64    96   128   160   192   224   256   288   320   352   384   416   448   480
.001 .325 .096 .198 .029 .014 .010 .010 .012 .003 .003 .005 .003 .003 .002

  512   544   576 1024 1536 2048 2560 3072 3584 4096 4608
.004 .005 .009 .043 .217 .000 .000 .000 .000 .000 .000

IP Flow Switching Cache, 4456704 bytes

65527 active, 9 inactive, 2364260060 added

4143879566 aging polls, 0 flow alloc failures

Active flow timeout in 30 minutes
Inactive flow 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-Telnet    1398292      0.3      14    156      4.6      6.0      17.2

TCP-FTP       9956998      23.1        1     41     24.2      0.0      4.8

TCP-FTPD      185530      0.0        1     66      0.0      1.5      17.4

TCP-WWW       440235639    102.5        8    483    919.5      2.9     10.1

TCP-SMTP      18951357      4.4        21   629    94.1      6.4     20.0

TCP-X         11340      0.0        1     48      0.0      0.2     40.8

TCP-BGP       4018      0.0        2     51      0.0      7.5     12.5

TCP-NNTP      2701390      0.6       104   846    65.5     10.6     16.9

TCP-Frag      38932      0.0        11   407      0.1      1.9     17.2

TCP-other     403434143    93.9        7    444   688.2      6.9     18.6

```

Cisco 7200 NetFlow Example (Cont.)

```

T C P - o t h e r      4 0 3 4 3 4 1 4 3      9 3 . 9      7      4 4 4      6 8 8 . 2      6 . 9      1 8 . 6

U D P - D N S          6 5 5 9 0 2 1 4      1 5 . 2      1      1 1 4      2 4 . 0      1 . 6      1 7 . 7

U D P - N T P          2 4 1 5 6 0 0      0 . 5      1      7 6      0 . 6      0 . 5      1 8 . 6

U D P - T F T P        7 0 0 1 1      0 . 0      5      7 7      0 . 0      3 2 . 2      1 7 . 8

U D P - F r a g        1 0 1 7 5 8 2      0 . 2      8 5      8 8      2 0 . 1      1 4 . 4      1 7 . 9

U D P - o t h e r      4 6 2 3 7 5 8 3 4      1 0 7 . 6      1 1      3 9 2      1 1 8 9 . 0      5 . 3      2 3 . 5

I C M P                8 5 6 3 2 3 2 5 1      1 9 9 . 3      1      8 9      2 1 7 . 4      0 . 3      3 7 . 7

I G M P                9 8      0 . 0      4 2 7 5      4 4 4      0 . 0      4 8 7 . 8      1 5 . 4

I P I N I P            4 6      0 . 0      1 1 2 2 9      4 1 2      0 . 1      1 0 3 9 . 7      6 . 8

G R E                  1 0 4 6 4 3      0 . 0      1 0      8 6      0 . 2      4 7 . 9      1 5 . 8

I P - o t h e r        9 7 6 6 6 2 7      2 . 2      1 0 2      3 1 8      2 3 2 . 5      8 5 . 6      1 9 . 7

T o t a l !:          2 3 6 4 1 9 4 5 3 3      5 5 0 . 4      6      4 1 1      3 4 8 1 . 2      3 . 3      2 4 . 3

```

Cisco 7200 NetFlow Example (Cont.)

Src If	Src IP address	Dst If	Dst IP address	Pr	Src P	Dst P	Pkts
Fa 0 / 1	1 0 . 6 6 . 7 4 . 4 6	Fa 0 / 0	2 1 9 . 1 0 3 . 1 2 9 . 1 6 2	0 1	0 0 0 0	0 8 0 0	1
Fa 0 / 1	1 0 . 6 6 . 1 1 5 . 1 8 2	Fa 0 / 0	1 9 4 . 2 2 . 1 1 4 . 1 9 8	0 1	0 0 0 0	0 8 0 0	1
Fa 2 / 1	1 0 . 6 6 . 7 4 . 4 6	Fa 0 / 0	6 1 . 7 9 . 2 2 7 . 1 2 3	0 1	0 0 0 0	0 8 0 0	1
Fa 0 / 1	1 0 . 6 6 . 7 4 . 4 6	Fa 0 / 0	2 1 1 . 1 6 7 . 1 0 5 . 2 4 2	0 1	0 0 0 0	0 8 0 0	1
Fa 0 / 0	1 2 9 . 4 2 . 1 8 4 . 3 5	N u l l	6 4 . 1 0 4 . 1 9 3 . 1 9 8	0 6	2 8 9 1	0 0 1 9	3
Fa 2 / 1	1 0 . 6 6 . 1 1 5 . 1 8 2	Fa 0 / 0	2 0 2 . 2 0 . 1 3 8 . 1 8 4	0 1	0 0 0 0	0 8 0 0	1
Fa 2 / 1	1 0 . 6 6 . 1 1 5 . 1 8 2	Fa 0 / 0	6 3 . 7 6 . 2 3 7 . 2 5 5	0 1	0 0 0 0	0 8 0 0	1
Fa 2 / 1	1 0 . 6 6 . 7 4 . 4 6	Fa 0 / 0	6 1 . 2 0 5 . 2 1 4 . 4 5	0 1	0 0 0 0	0 8 0 0	1
Fa 2 / 1	1 0 . 6 6 . 1 1 5 . 1 8 2	Fa 0 / 0	2 2 0 . 1 1 4 . 1 5 7 . 1	0 1	0 0 0 0	0 8 0 0	1
Fa 0 / 0	6 4 . 1 0 4 . 2 5 2 . 1 9 6	Fa 2 / 1	6 4 . 1 0 4 . 2 0 0 . 2 1 0	1 1	0 0 0 0	0 0 0 0	1
Fa 0 / 1	6 4 . 1 0 4 . 1 9 2 . 1 3 0	Fa 0 / 0	2 1 7 . 1 3 6 . 1 9 . 1 0 3	1 1	2 7 1 0	2 7 1 0	3 6 0 3

Tracking TOS with NetFlow

```

7 2 0 0 -3 -N etFlow # show ip cache verbose flow

SrcIf          SrcIPaddress      DstIf          DstIPaddress      PrTOS  Flgs  Pkts
Port Msk A S      Port Msk A S      NextHop          B / Pk  Active

SR6 / 0          2 1 0 .2 1 0 .2 1 0 .2      PO1 / 0          2 0 0 .2 0 0 .2 0 0 .2      FF 0 0      1 0      2 1 K
0 0 0 0 / 0      0      0 0 0 0 / 0      0      0 .0 .0 .0          1 4 9 6      6 6 5 .4

SR6 / 0          2 1 0 .2 1 0 .2 1 0 .2      PO1 / 0          2 0 0 .2 0 0 .2 0 0 .2      0 6 C 0      0 0      2 1 K
0 0 0 0 / 0      0      0 0 0 0 / 0      0      0 .0 .0 .0          1 4 9 6      6 6 6 .0

7 2 0 0 -3 -N etFlow # show ip cache verbose flow

SrcIf          SrcIPaddress      DstIf          DstIPaddress      PrTOS  Flgs  Pkts
Port Msk A S      Port Msk A S      NextHop          B / Pk  Active

Et1 / 1          5 2 .5 2 .5 2 .1          Fd4 / 0          4 2 .4 2 .4 2 .1          0 1 5 5      1 0      3 7 4 8
0 0 0 0 / 8      5 0          0 0 0 0 / 8      4 0          2 0 2 .1 2 0 .1 3 0 .2          2 8      1 7 .8

Et1 / 2          5 2 .5 2 .5 2 .1          Fd4 / 0          4 2 .4 2 .4 2 .1          0 1 C C      1 0      3 5 6 8
0 0 0 0 / 8      5 0          0 0 0 0 / 8      4 0          2 0 2 .1 2 0 .1 3 0 .2          2 8      1 7 .8

Et1 / 2          1 0 .1 .3 .2          Fd4 / 0          4 2 .4 2 .4 2 .1          0 1 C 0      1 0      1 1 2 4
0 0 0 0 / 0      0          0 0 0 0 / 8      4 0          2 0 2 .1 2 0 .1 3 0 .2          2 8      1 7 .8

```

Hex	Decimal	Binary	
55	85	0 1 0 1 0 1 0 1	Precedence 2 - Immediate (Class 2), Delay - low, Reliability - high, Endpoints of transport protocol ECN - capable
C0	192	1 1 0 0 0 0 0 0	Precedence 6 - Internetwork Control (Routing Protocols)
CC	204	1 1 0 0 1 1 0 0	Precedence 6 - Internetwork Control (Routing Protocols), Throughput - high, Reliability - high

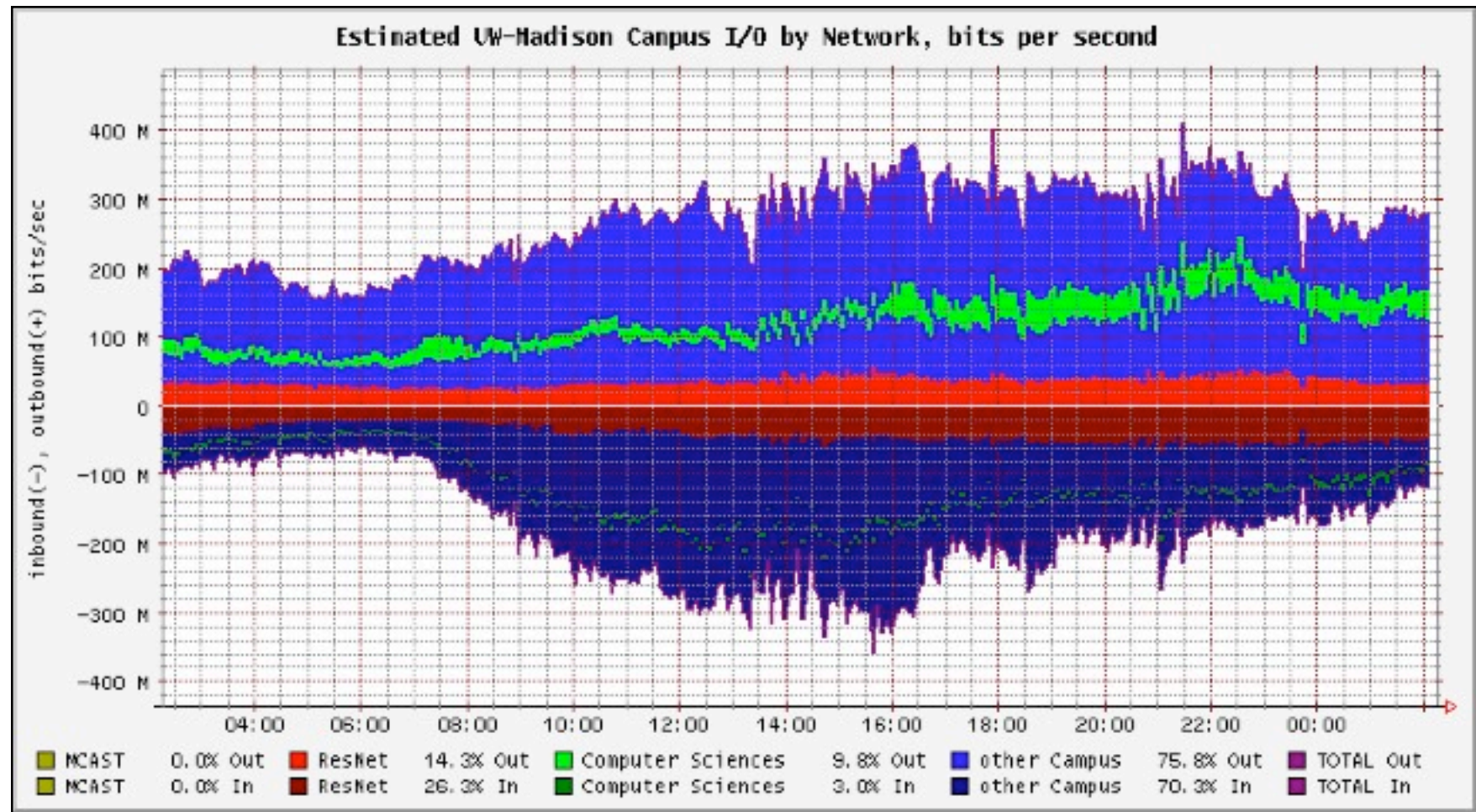
Getting Started with NetFlow Collection: The OSU Flow-Tools

- Open source NetFlow collection and retrieval tools
- Developed and maintained by Mark Fullmer, available from <http://www.splintered.net/sw/flow-tools/>
- Runs on common *NIX platforms (Linux, FreeBSD, Mac OS/X, Solaris, etc.)
- Command-line tools allow for very display/sorting of specific criteria (source/dest IP, source/dest ASN, protocol, port, etc.)
- Data can be batched and imported into database such as Oracle, MySQL, Postgres, etc.
- Can be combined with other tools to provide visualization of traffic patterns
- Many other useful features—check it out today

Getting Started with NetFlow Visualization: FlowScan

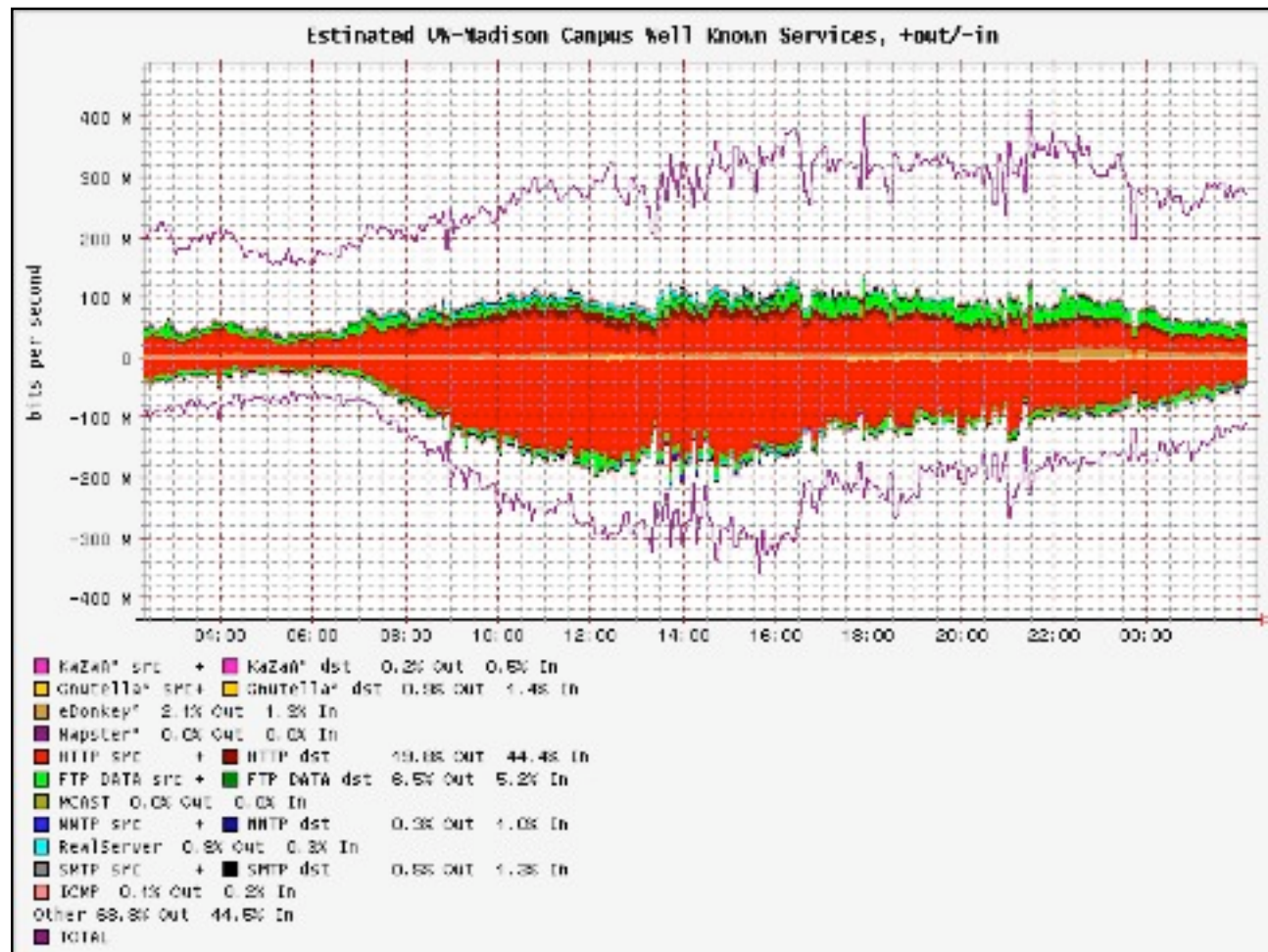
- Open source NetFlow graphing/visualization tools
- Developed and maintained by Dave Plonka, available from <http://net.doit.wisc.edu/~plonka/FlowScan/>
- Runs on common *NIX platforms (Linux, FreeBSD, Mac OS/X, Solaris, etc.)
- Makes use of NetFlow data collected via flow-tools to build traffic graphs
- Top-talkers by subnet, other types of reports supported
- Makes use of RRDTool for graphing
- Add-ons such as JKFlow module allow more detailed graphing

Example: FlowScan Graphs



Source: University of Wisconsin

Example: FlowScan Graphs



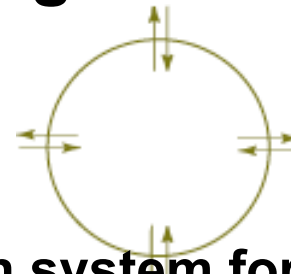
Source: University of Wisconsin

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**

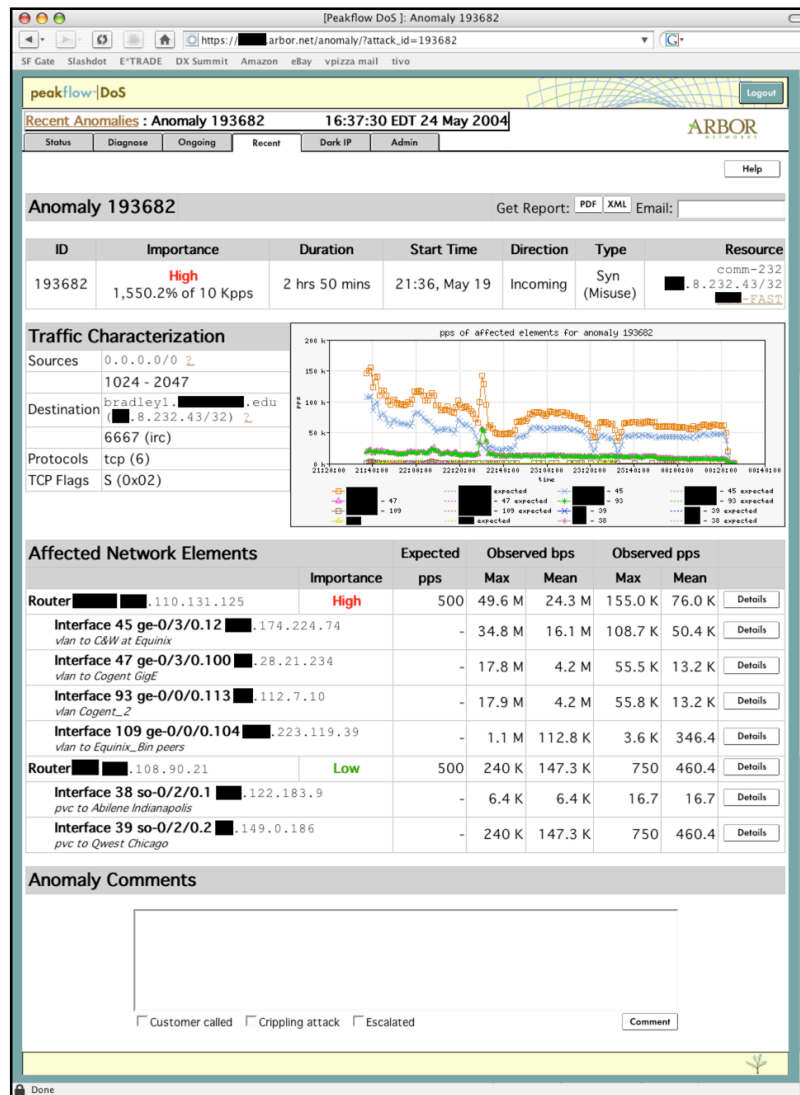
NetFlow-Based Traffic Characterization and Anomaly Detection with Arbor Networks

Network Anomaly Detection and Traffic Characterization/Capacity Planning

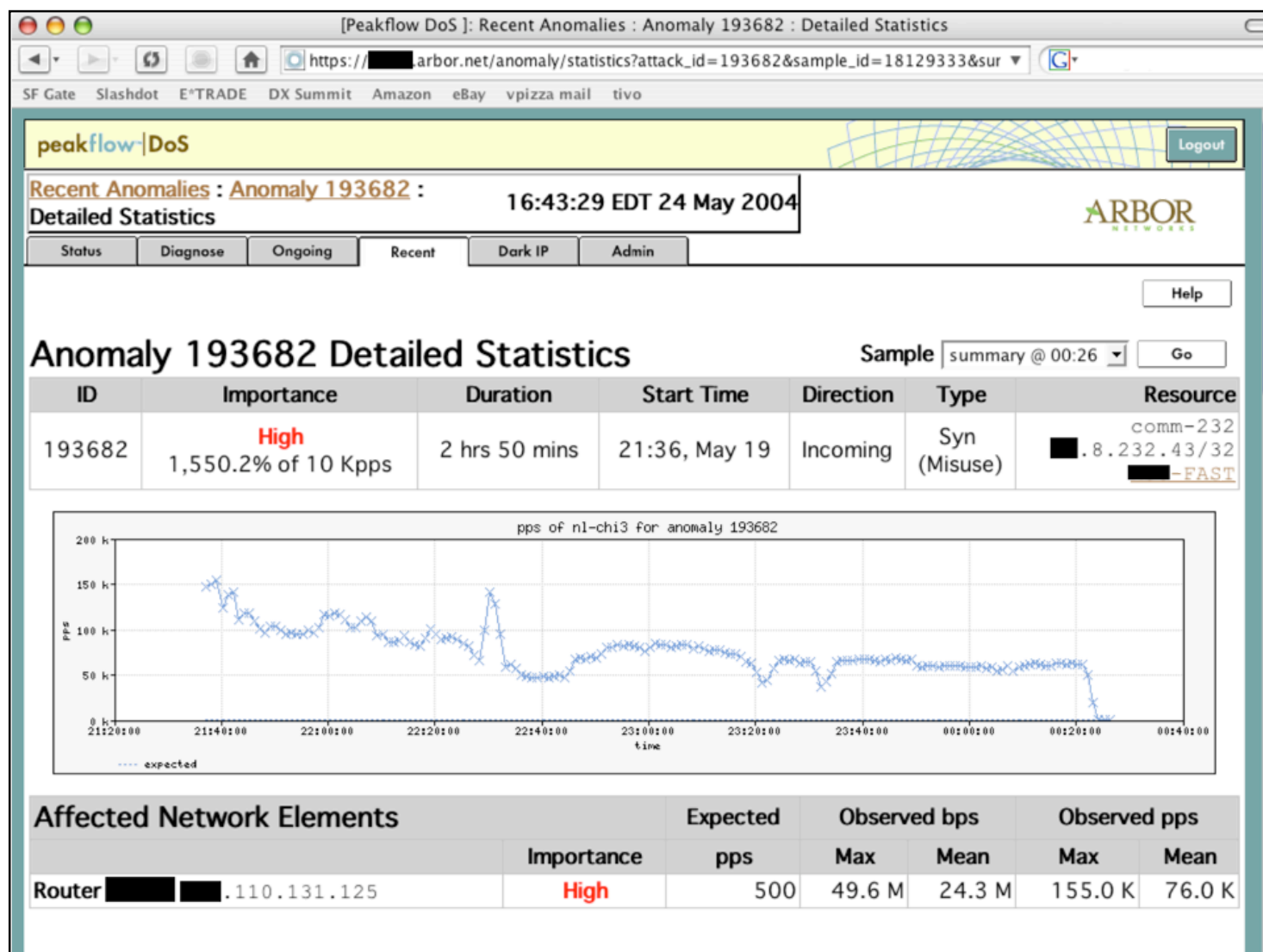


- Most widely deployed anomaly detection system for SPs
- Uses NetFlow to quickly identify, classify, and scope DoS, worms, etc.
- Traffic component combines NetFlow traffic characterization with BGP
- Allows comprehensive peering analysis in real-time
- A “force multiplier” which greatly reduces reaction-times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

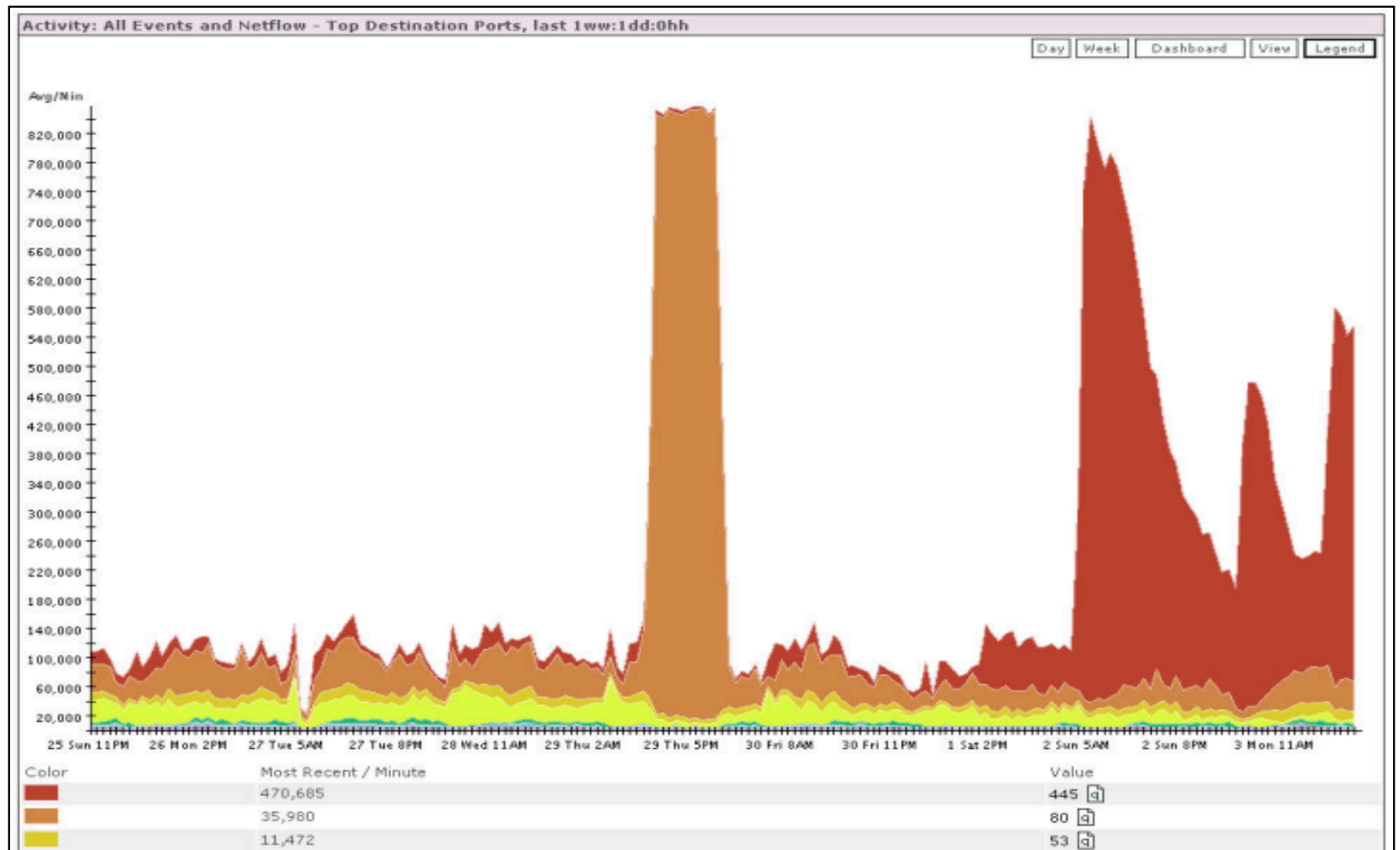
Anomaly Example



Anomaly Example: Detail



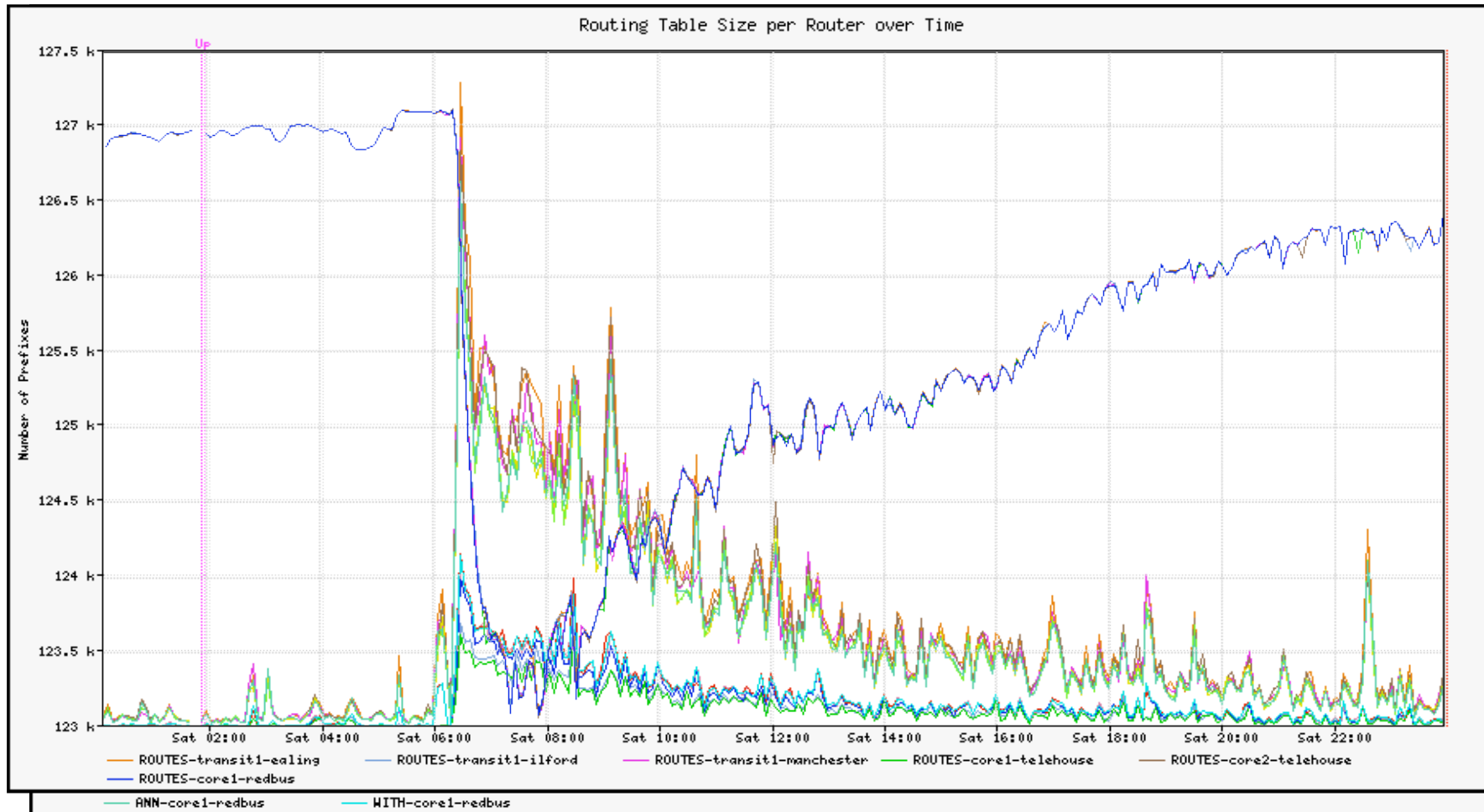
Sasser Detection



BGP: Why Do We Care?

- Large-scale network security events such as worms, DDoS attacks, etc., often produce side-effects visible in the global routing table
- Correlating BGP information with other forms of telemetry (NetFlow, SNMP, RMON, etc.) can be effective in determining the true impact of incidents
- Zebra (<http://www.zebra.org>) and Quagga (<http://www.quagga.net>) are two open source BGP daemons which can log BGP updates for further analysis
- Arbor PeakFlow SP Traffic provides BGP visualization, trending, NetFlow traffic correlation, additional functionality (http://www.arbornetworks.com/products_sp.php)
- RIBs/updates available from <http://archive.routeviews.org/> , <http://www.ripe.net/ris/index.html> , <http://www.renesys.com> (commercial, useful monitoring tools/services for your ASN)

BGP Example: SQL Slammer



NetFlow Layer 2 and Security Monitoring

- **Captures information from more Layer 3 fields:**
 - Time-to-Live**
 - Identification**
 - Fragment offset**
 - Packet length**
 - ID**
- **Captures information from Layer 2 fields**
 - Source MAC address from received frames**
 - Destination MAC address from xmitted frames**
 - VLAN ID from received and xmitted frames**

NetFlow Layer 2 and Security Monitoring: Benefits

- Added Layer 3 fields improve ability to ID DoS attacks
- New Layer 2 capability helps ID path DoS attacks take through network

Determine interface DoS attack is arriving on

MAC and VLAN ID fields show attack's previous hop

- Available in 12.3(14)T

http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123newft/123t/123t_14/nflwsec1.htm

Design Tips for NetFlow-Based Detection

- Use sampled NetFlow to reduce CPU usage on software platforms by up to **80%**
- Sampling rate is configurable
- Use sampled NetFlow for traffic capacity and network planning
- Agree on which fields of NetFlow to track
- Do not export versions 5, 7 and 9 simultaneously with version 8
- Plan NetFlow deployment in the network topology to avoid a design that creates duplicate flows for billing
- Use a dedicated interface/VLAN for NetFlow data export (NDE)
- Monitor lost packet counter in NFC
- Check the export link bandwidth
Estimate export of 1% to 1.5% of the interface throughput

Core Security Techniques



***Anycast* and Security**

Anycast and Security

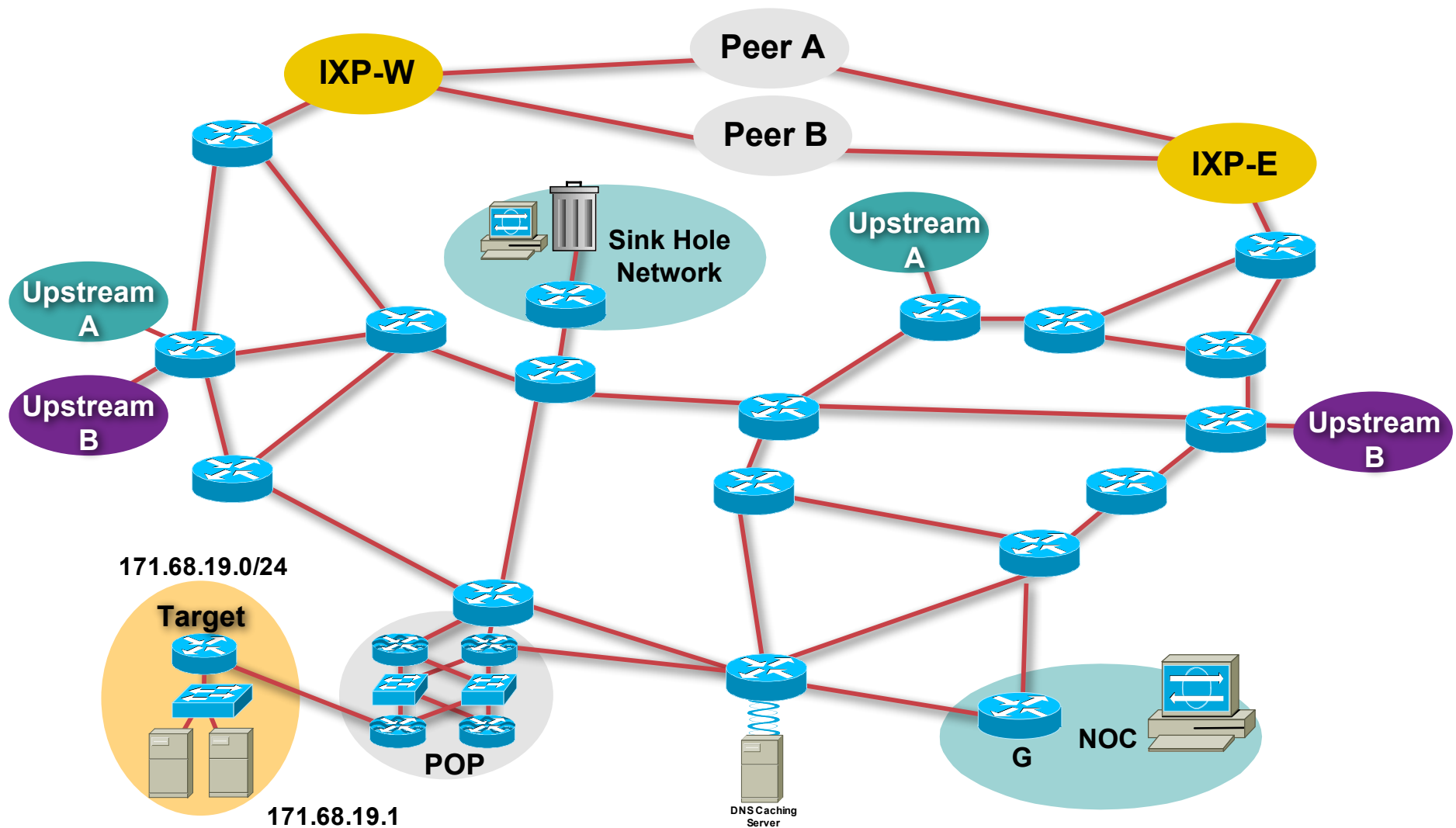
- **IPv4's Anycast technique can be used as a security tool.**

Provides topological separation. Making it harder to attack a service (DNS, AAA, etc).

Topological separation provides a means to put sink holes through out the network.

Two devices looking like one offers a way to have customer iBGP origination points to be two routers vs one without the added IGP memory consumption.

What is Anycast?



What *isn't* Anycast?

- **Not a protocol, not a different version of IP, nobody's proprietary technology.**
- **Doesn't require any special capabilities in the servers, clients, or network.**
- **Doesn't break or confuse existing infrastructure.**

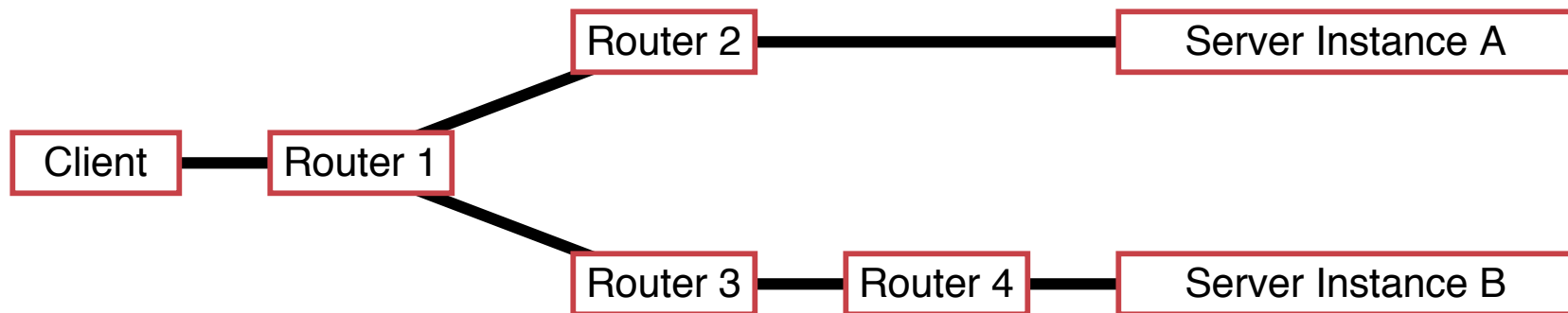
What *is* Anycast?

- Just a configuration methodology.
- Mentioned, although not described in detail, in numerous RFCs since time immemorial.
- It's been the basis for large-scale content-distribution networks since at least 1995.
- It's gradually taking over the core of the DNS infrastructure, as well as much of the periphery of the world wide web.

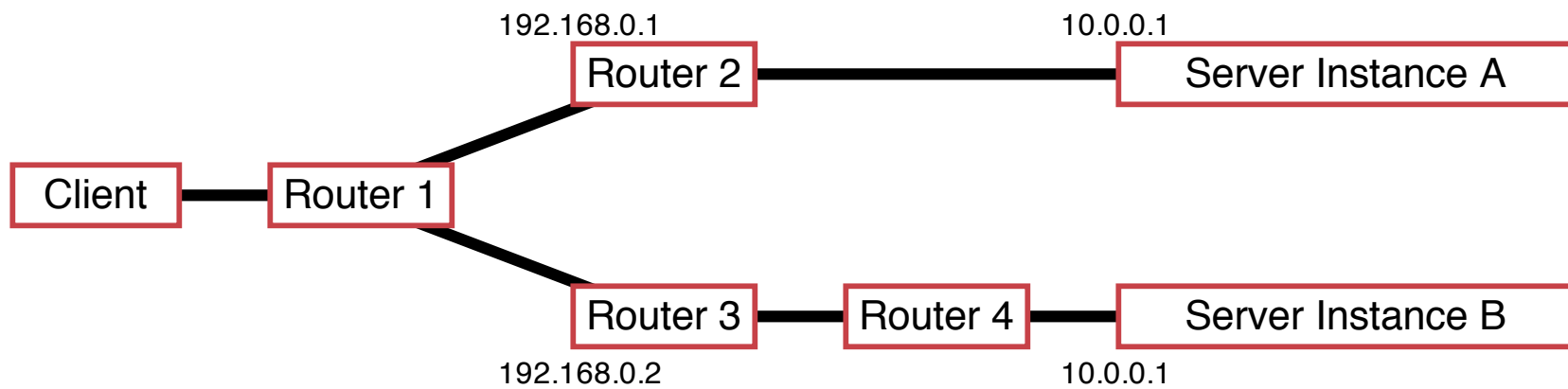
How Does Anycast Work?

- **The basic idea is extremely simple:**
- **Multiple instances of a service share the same IP address.**
- **The routing infrastructure directs any packet to the topologically nearest instance of the service.**
- **What little complexity exists is in the optional details.**

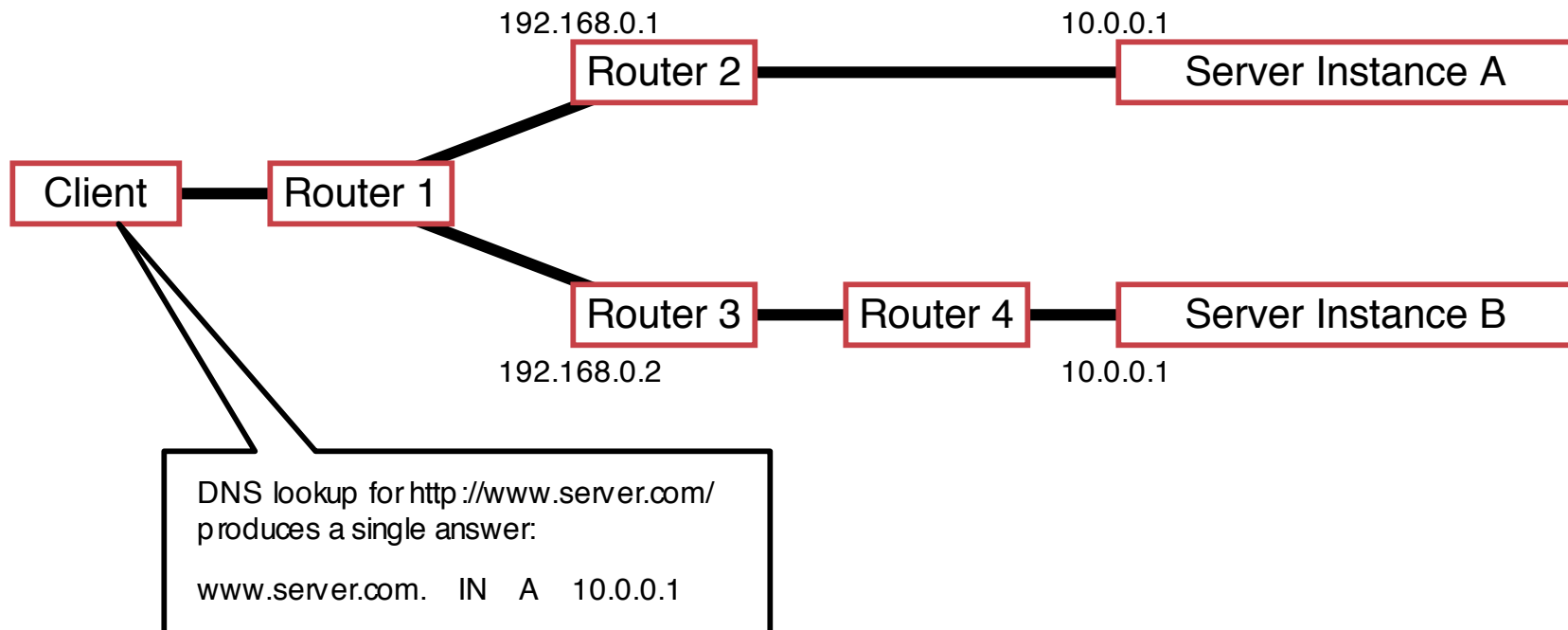
Example



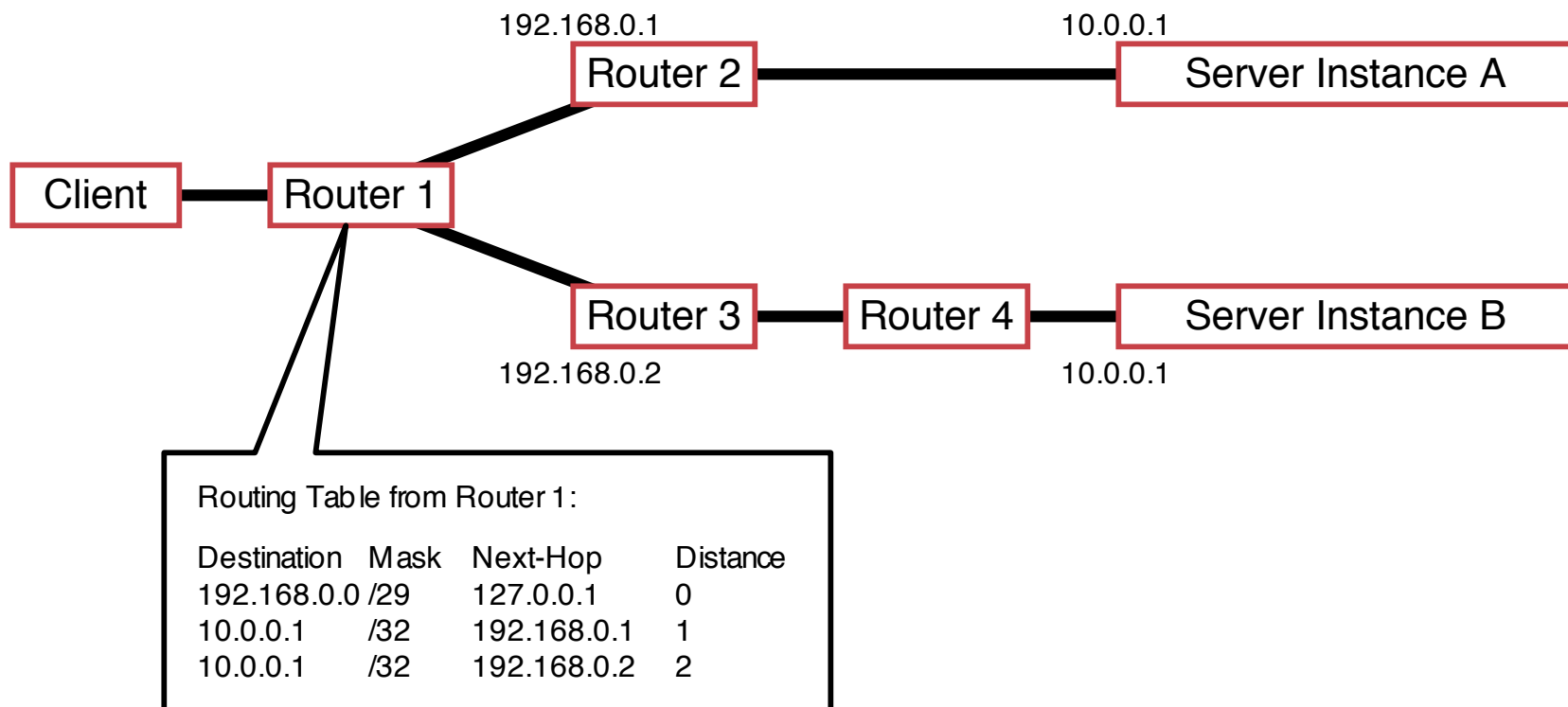
Example



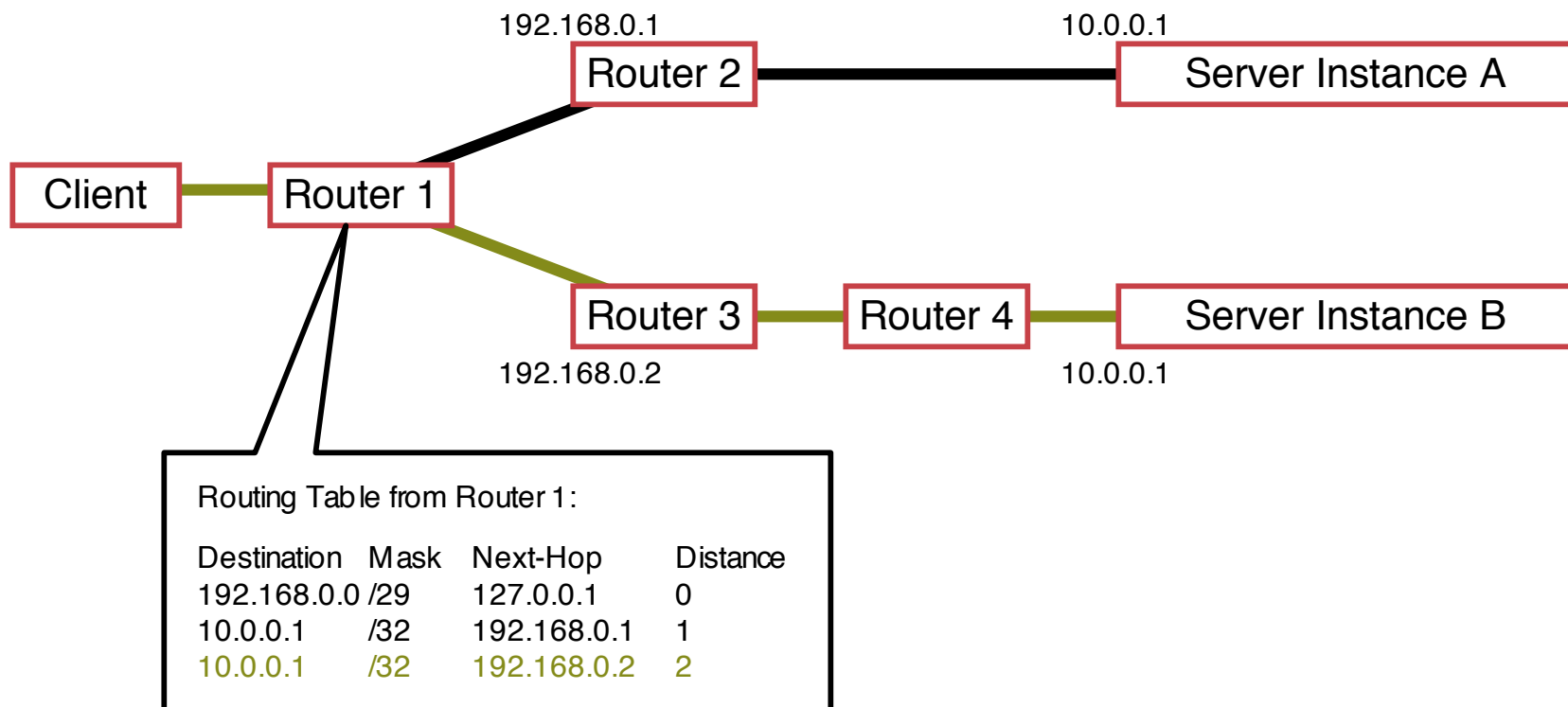
Example



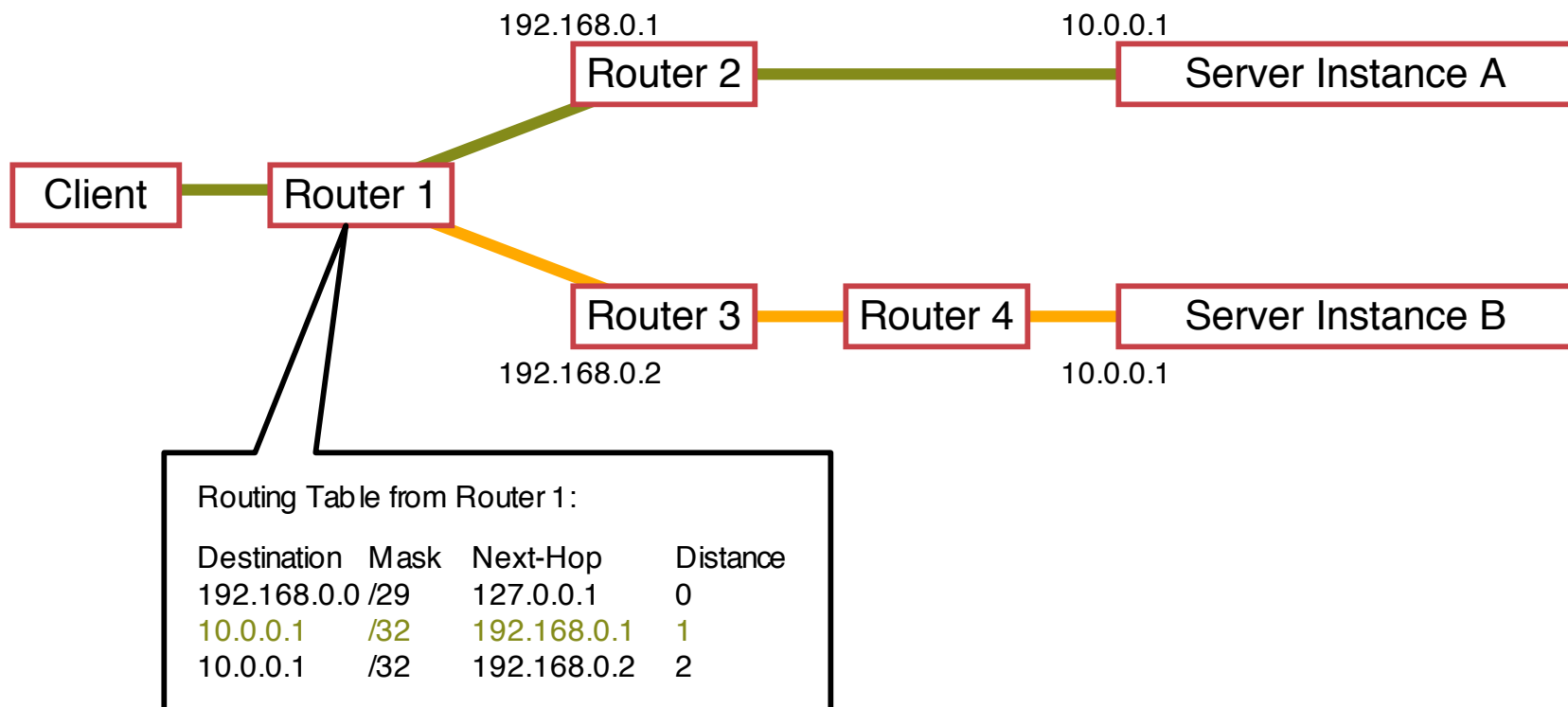
Example



Example

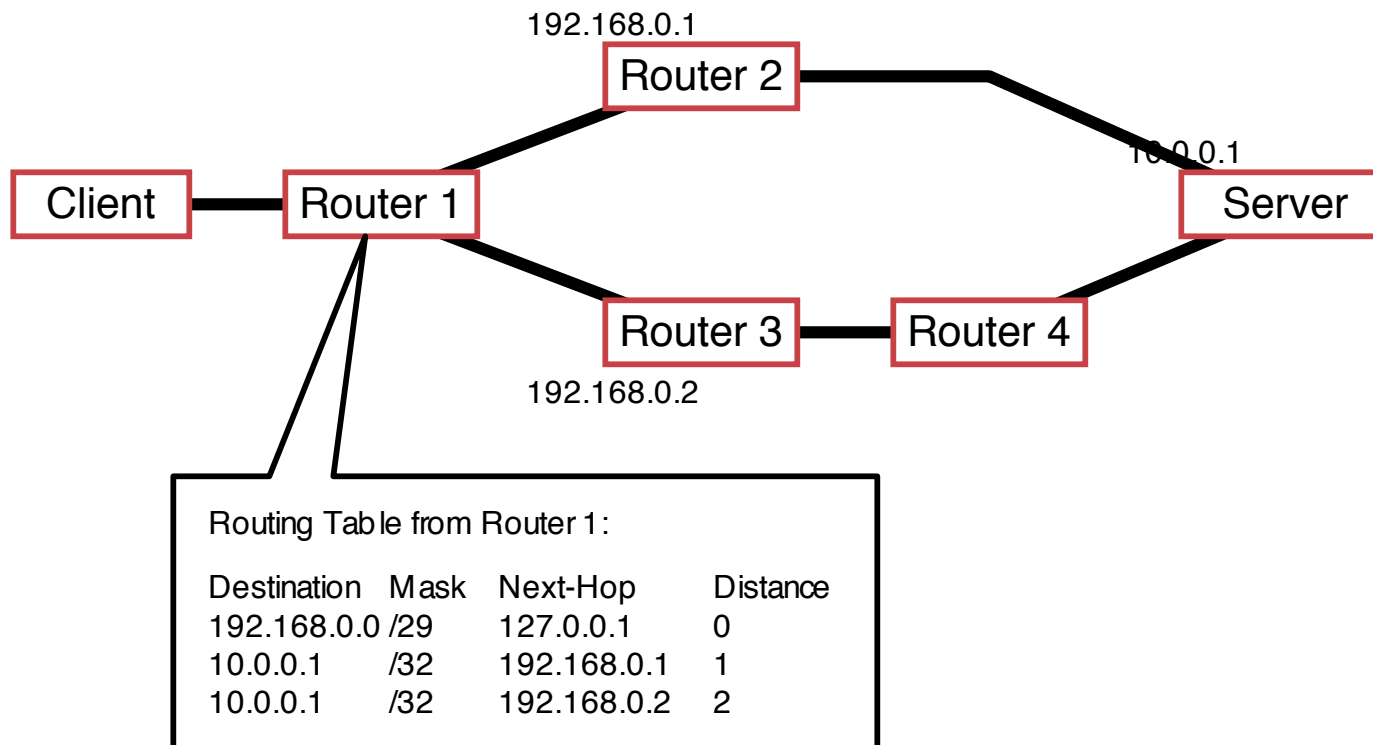


Example



Example

What the routers think the topology looks like:



Building an Anycast Server Cluster

- **Anycast can be used in building either local server clusters, or global networks, or global networks of clusters, combining both scales.**
- **F-root is a local anycast server cluster, for instance.**

Caveats and Failure Modes

- **DNS resolution fail-over**
- **Long-lived connection-oriented flows**
- **Identifying which server is giving an end-user trouble**

DNS Resolution Fail-Over

- **In the event of poor performance from a server, DNS servers will fail over to the next server in a list.**
- **If both servers are in fact hosted in the same anycast cloud, the resolver will wind up talking to the same instance again.**
- **Best practices for anycast DNS server operations indicate a need for two separate overlapping clouds of anycast servers.**

Long-Lived Connection-Oriented Flows

- Long-lived flows, typically TCP file-transfers or interactive logins, may occasionally be more stable than the underlying Internet topology.
- If the underlying topology changes sufficiently during the life of an individual flow, packets could be redirected to a different server instance, which would not have proper TCP state, and would reset the connection.
- This is not a problem with web servers unless they're maintaining stateful per-session information about end-users, rather than embedding it in URLs or cookies.
- Web servers HTTP redirect to their unique address whenever they need to enter a stateful mode.
- Limited operational data shows underlying instability to be on the order of one flow per ten thousand per hour of duration.

Identifying Problematic Server Instances

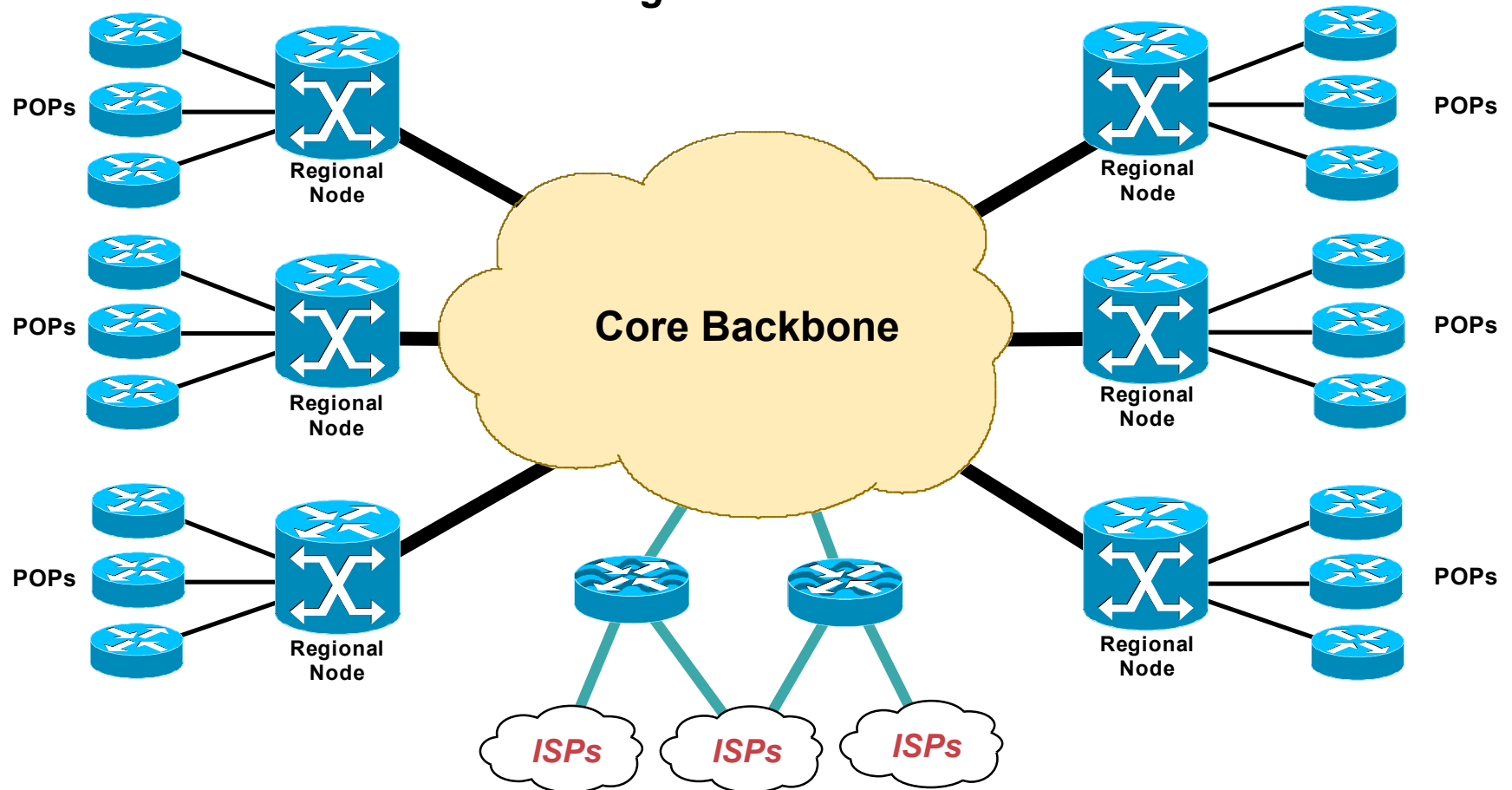
- **Some protocols may not include an easy in-band method of identifying the server which persists beyond the duration of the connection.**
- **Traceroute always identifies the *current* server instance, but end-users may not even have traceroute.**

A Security Ramification

- **Anycast server clouds have the useful property of sinking DOS attacks at the instance nearest to the source of the attack, leaving all other instances unaffected.**
- **This is still of some utility even when DOS sources are widely distributed.**

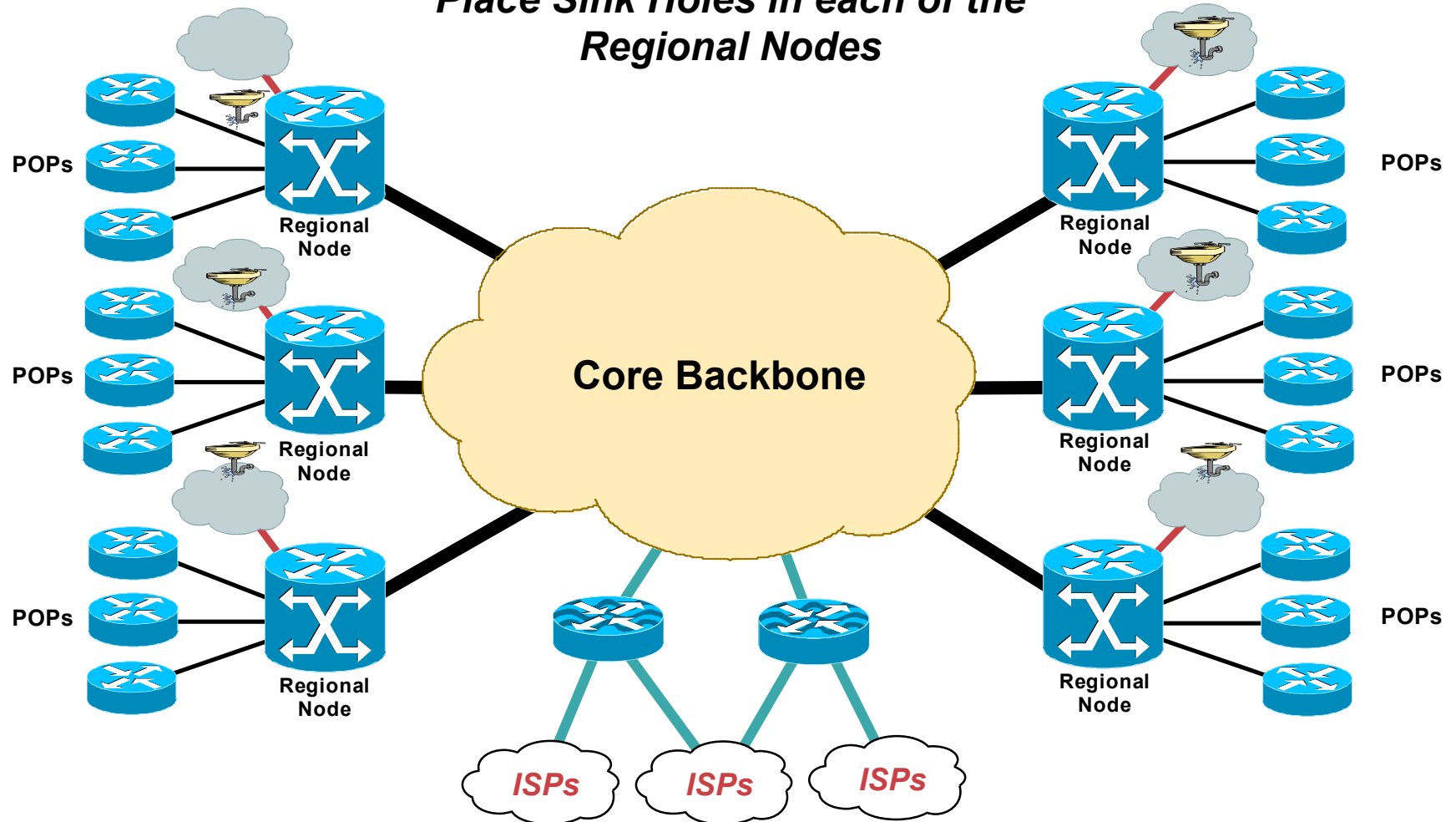
Anycast Sink Holes Example

*Template Backbone with
Regional Centers*



Anycast Sink Hole Placement

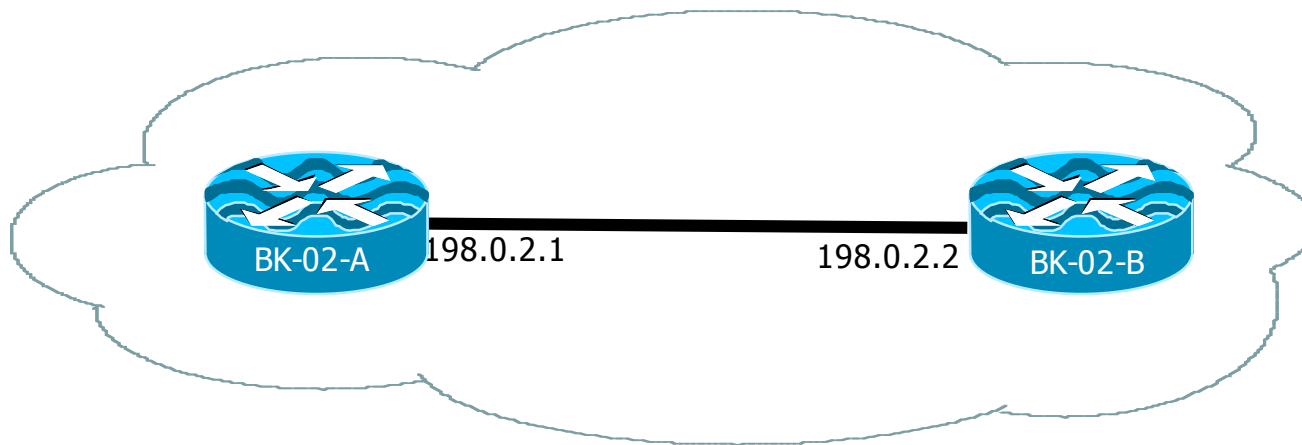
*Place Sink Holes in each of the
Regional Nodes*



Using Sink Holes to Protect Infrastructure Point to Point Links

Protecting the Backbone Point to Point Addresses

- Do you really need to reach the Backbone router's Point to Point Address from any router other than a directly connected neighbor?



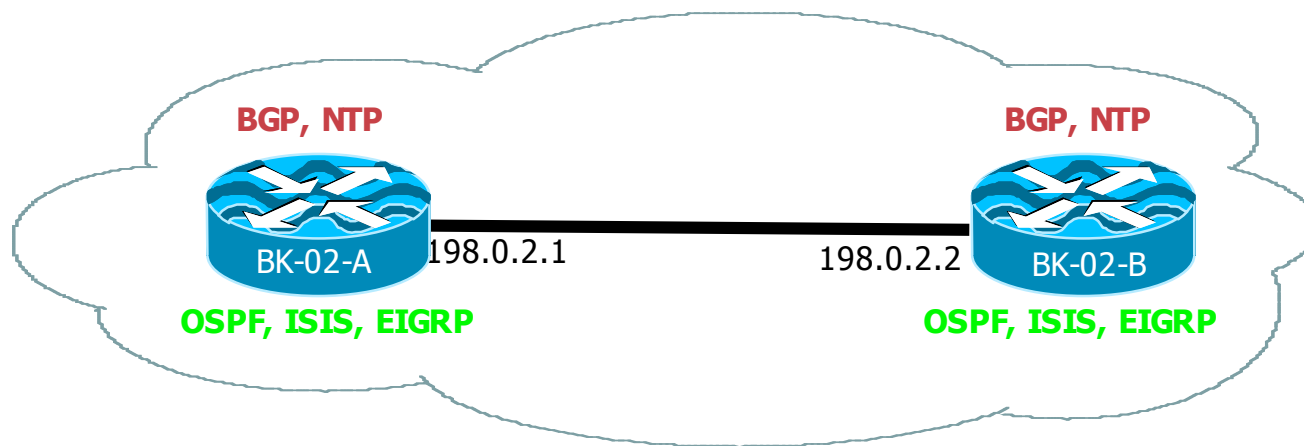
Protecting the Backbone Point to Point Addresses

- What could break?

Routing protocols are either loopback (BGP or NTP) or adjacent (OSPF, IS-IS, EIGRP).

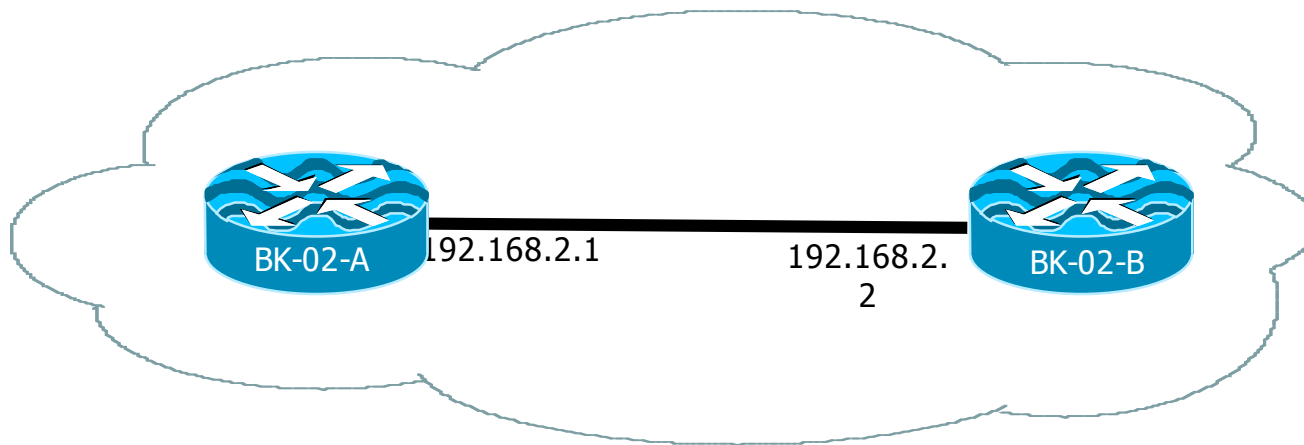
NOC can Ping the Loopback.

Traceroutes reply with the address in the reply.
Reachability of the source is not required.



Protecting the Backbone Point to Point Addresses

- **What have people done in the past:**
 - ACLs – Long term ACL management problems.**
 - RFC 1918 – Works – against the theme of the RFC – Traceroute still replies with RFC 1918 source address.**
 - Does not protect against a reflection attack.**



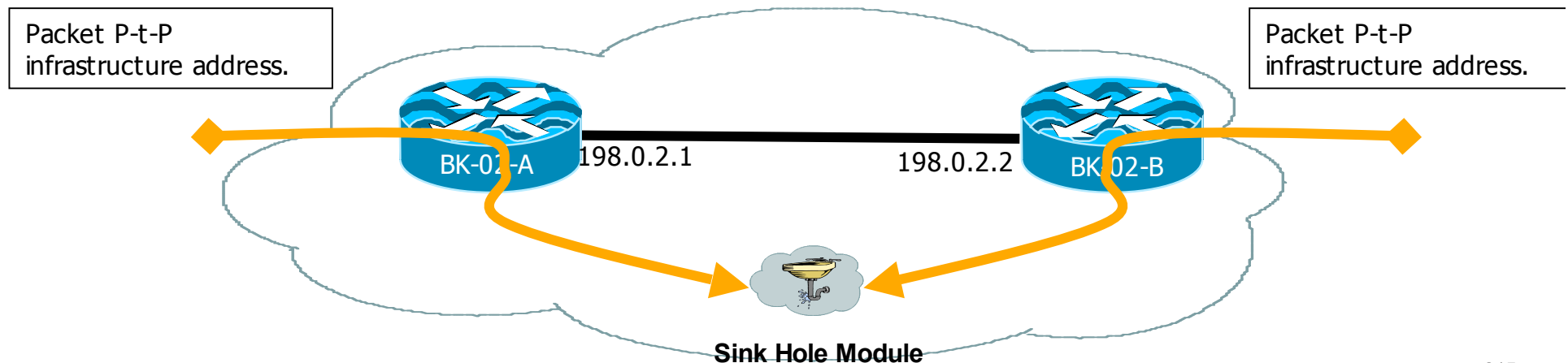
Protecting the Backbone Point to Point Addresses

- **Move the Point to Point Addresses blocks to IGP based Sink Holes.**

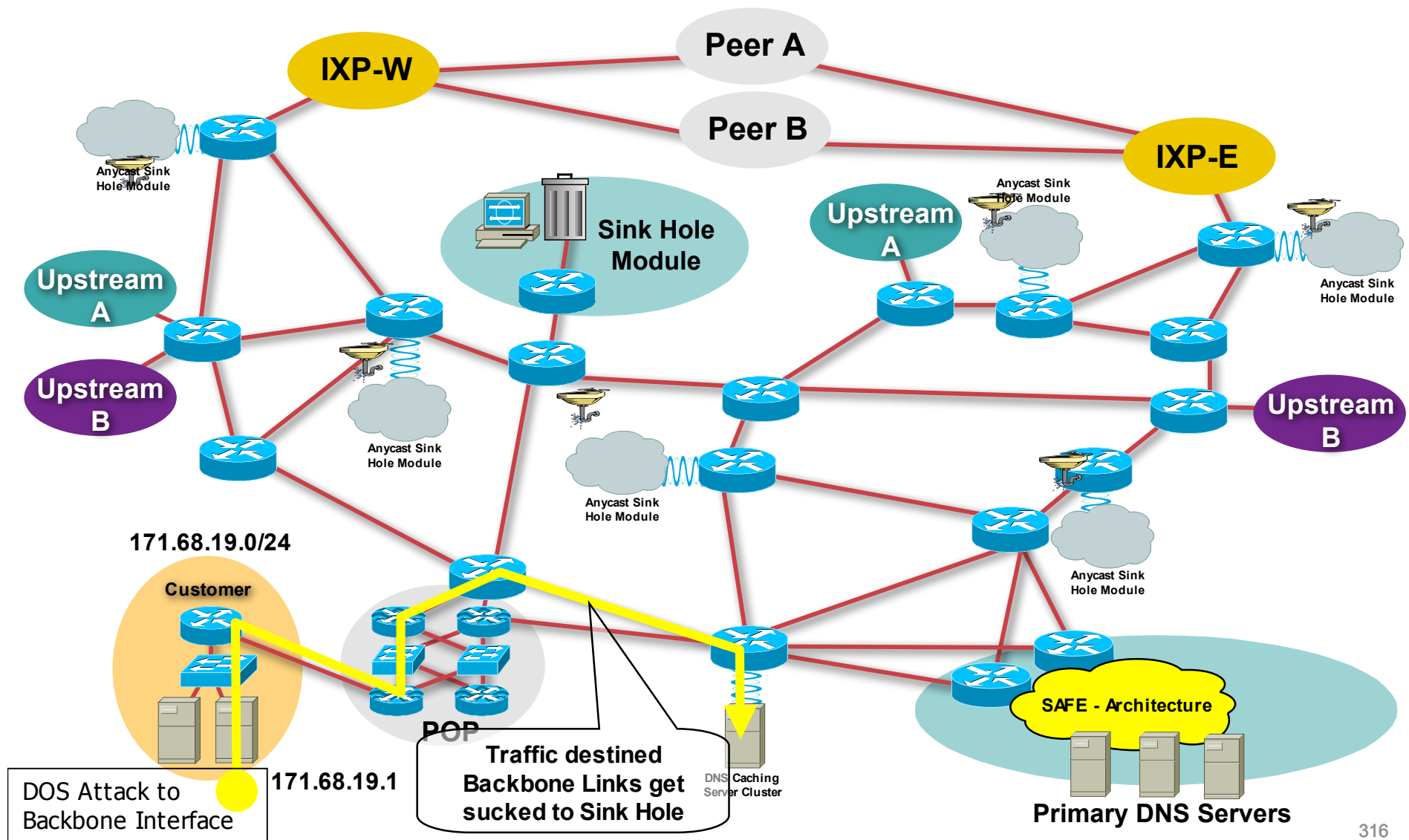
All packets to these addresses will be pulled into the Sink Hole.

People who could find targets with traceroute cannot now hit the router with an attack based on that intelligence.

Protects against internal and reflection based attacks.



Protecting the Backbone Point to Point Addresses



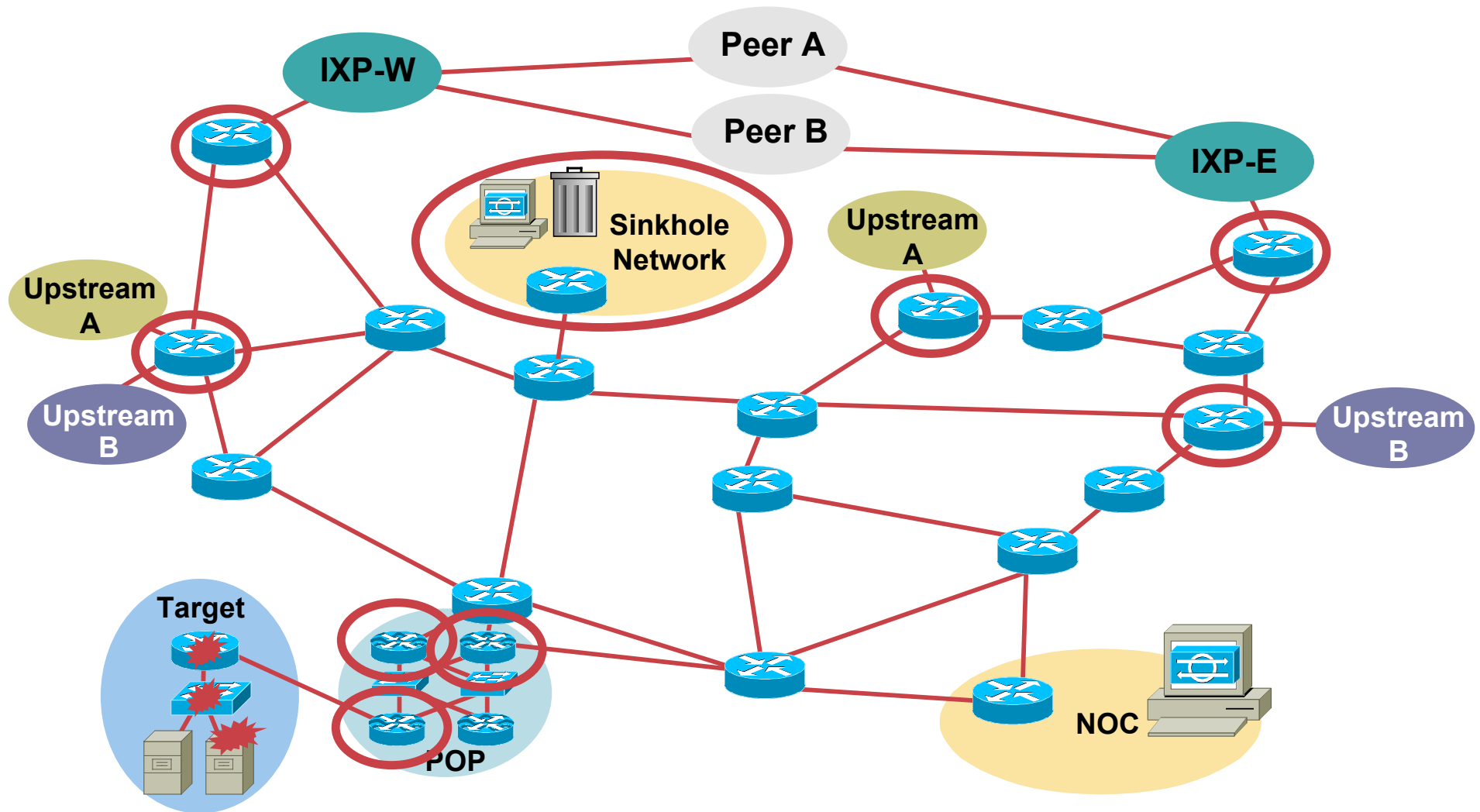
Reacting to Attacks



Reaction Tools

- **Wide range of response options exists**
 - Access-control lists**
 - QoS tools such as CAR, traffic policing and NBAR**
 - Firewalls**
 - Various IDS technologies: NIDS, HIDS, anomaly detection**
 - BGP triggers**
 - Packet scrubbing**
- **Today, we will focus on core-centric tools**

Where to React?



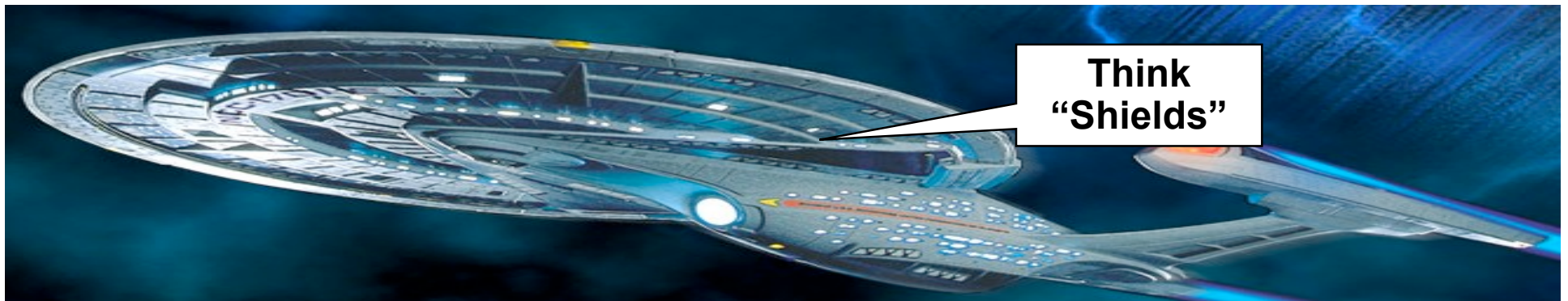
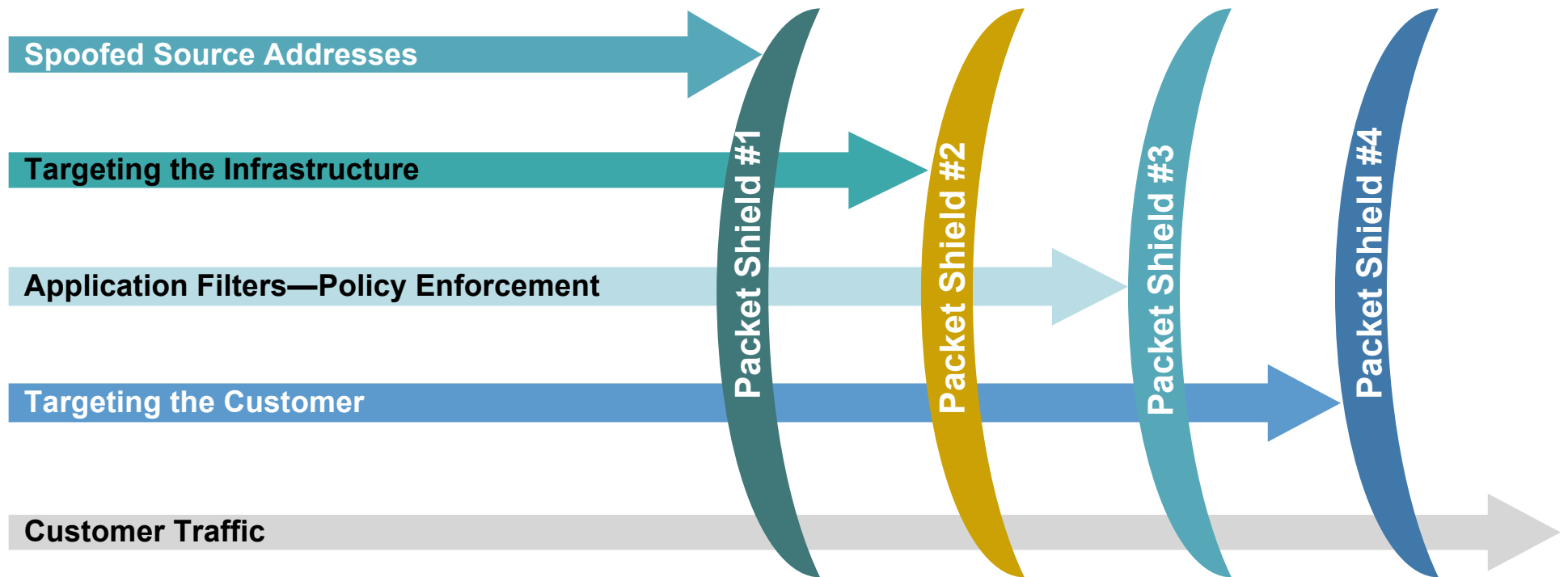
Reacting with ACLs



Reacting to an Attack with ACLs—Changed

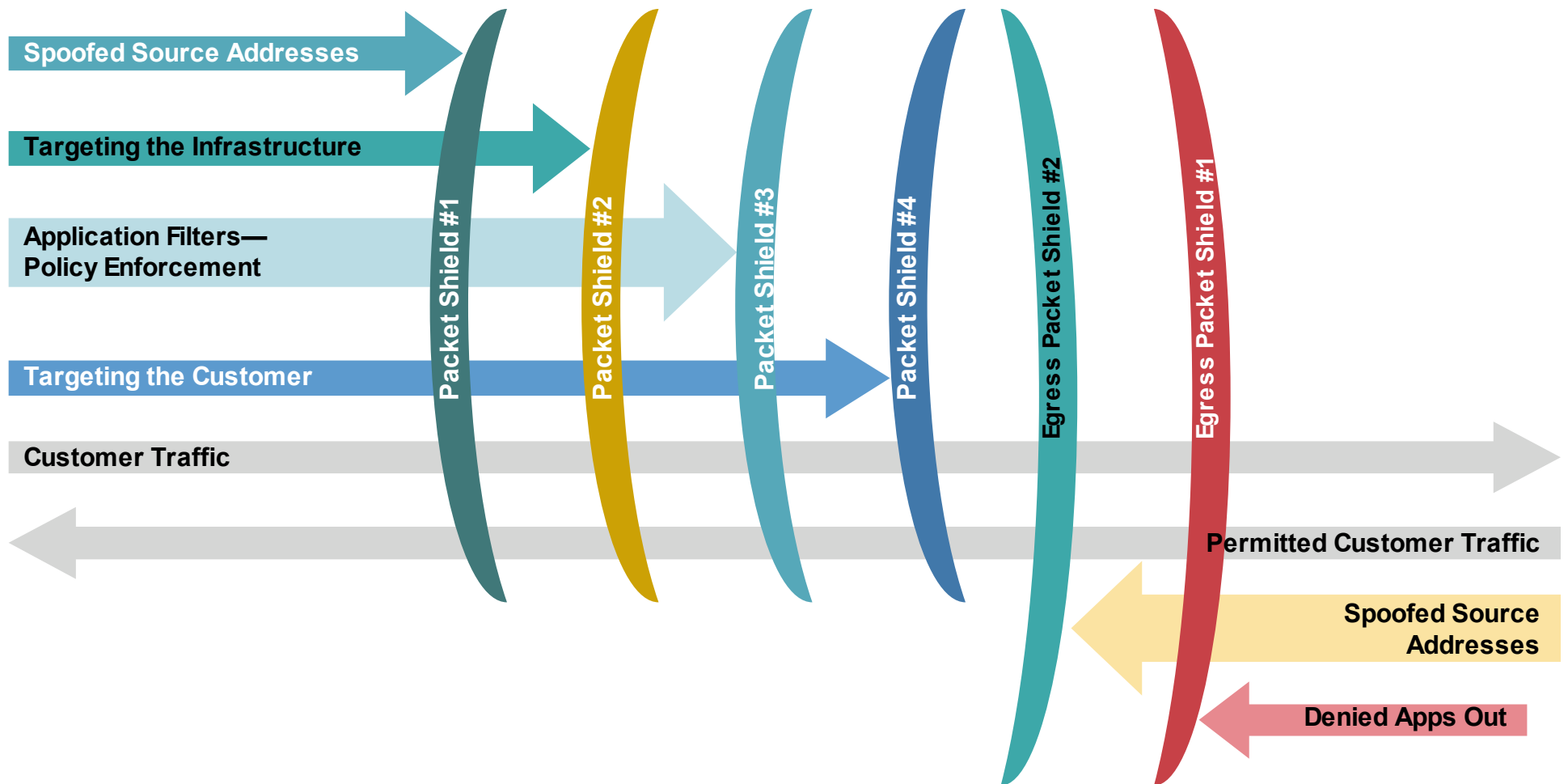
- **Traditional method for stopping attacks**
- **Scaling issues encountered:**
 - Operational difficulties**
 - Changes on the fly**
 - Multiple ACLs per interface**
 - Performance concerns**
- **How does the ACL load into the router? Does it interrupt packet flow?**
- **How many ACEs can be supported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

Packet Filtering Viewed Horizontally



Packet Filtering

Remember to Filter the Return Path



ACL Construction

- **Most common problem: poorly-constructed ACLs**
- **Scaling and maintainability issues with ACLs are commonplace**
- **Make your ACLs as modular and simple as possible: KISS**
- **Examples and best practices see:**

Transit Access Control Lists: Filtering at Your Edge

http://www.cisco.com/en/US/tech/tk648/tk361/technologies_white_paper09186a00801afc76.shtml

Protecting Your Core: Infrastructure Protection ACLs

http://www.cisco.com/en/US/tech/tk648/tk361/technologies_white_paper09186a00801a1a55.shtml

ACL Categories

Explicit Permit	Explicit Deny
Anti-Spoofing	Anti-Spoofing
Anti-Bogon (Source)	Anti-Bogon (Source)
Infrastructure	Infrastructure
Incident Reaction	Incident Reaction
Explicit Permit L3	Explicit Deny L3
Explicit Permit L4	Explicit Deny L4
Explicit Deny Everything Else (Auditing)	Explicit Permit Everything Else (Auditing)

ACL Categories: Hybrid Philosophy

Hybrid Permit/Deny

- Anti-spoofing
 - Anti-bogon (source)
 - Infrastructure
 - Explicit deny specific L3
 - Explicit deny specific L4
 - Incident reaction
 - Explicit permit L3 (good traffic)
 - Explicit permit L4 (good traffic)
 - Explicit deny everything else (auditing)

ACL Maintenance: Frequency of Change

Hybrid Permit/Deny

• Anti-spoofing	Rarely Changes
• Anti-bogon (source)	Rarely Changes
• Infrastructure	Rarely Changes
• Explicit deny specific L3	Sometimes Changes
• Explicit deny specific L4	Sometimes Changes
• Incident reaction	Changes Everyday
• Explicit permit L3 (good traffic)	Sometimes Changes
• Explicit permit L4 (good traffic)	Sometimes Changes
• Explicit deny everything else (auditing)	Rarely Changes

ACL Links

- **ACL and QoS TCAM Exhaustion Avoidance on Cisco Catalyst 4500 Switches**

http://www.cisco.com/en/US/products/hw/switches/ps663/products_tech_note09186a008054a499.shtml

- **Understanding ACL on Cisco Catalyst 6500 Series Switches**

http://www.cisco.com/en/US/products/hw/switches/ps708/products_white_paper09186a00800c9470.shtml

- **Implementing Access Lists on Cisco 12000 Series Internet Routers**

http://www.cisco.com/warp/public/63/acl_12000.html

- **3550 ACLs**

<http://www.cisco.com/univercd/cc/td/doc/product/lan/c3550/12225seb/scg/swacl.htm>

- **Access Control Lists and IP Fragments**

http://www.cisco.com/warp/public/105/acl_wp.html

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

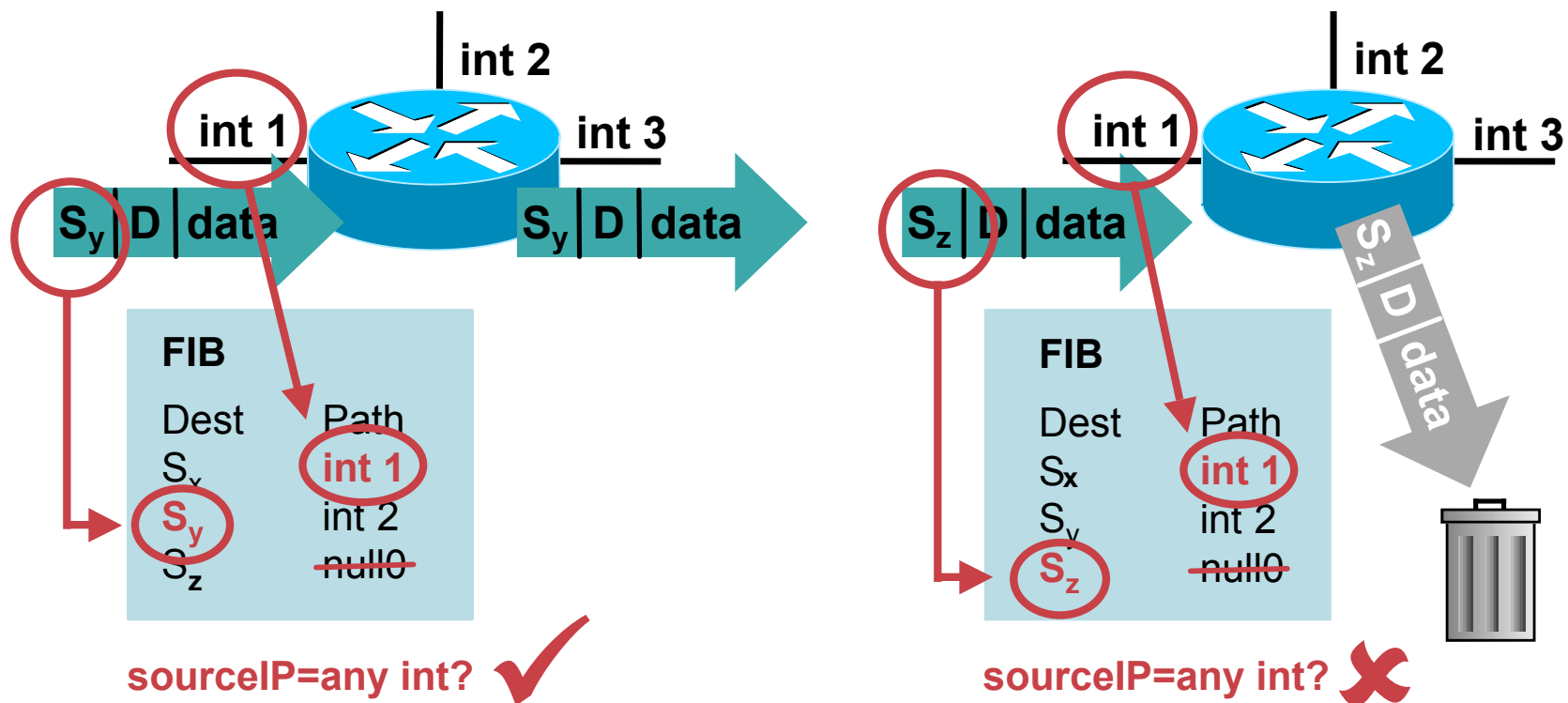


Flipping RTBH Around: Triggered Source Drops

- **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



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?

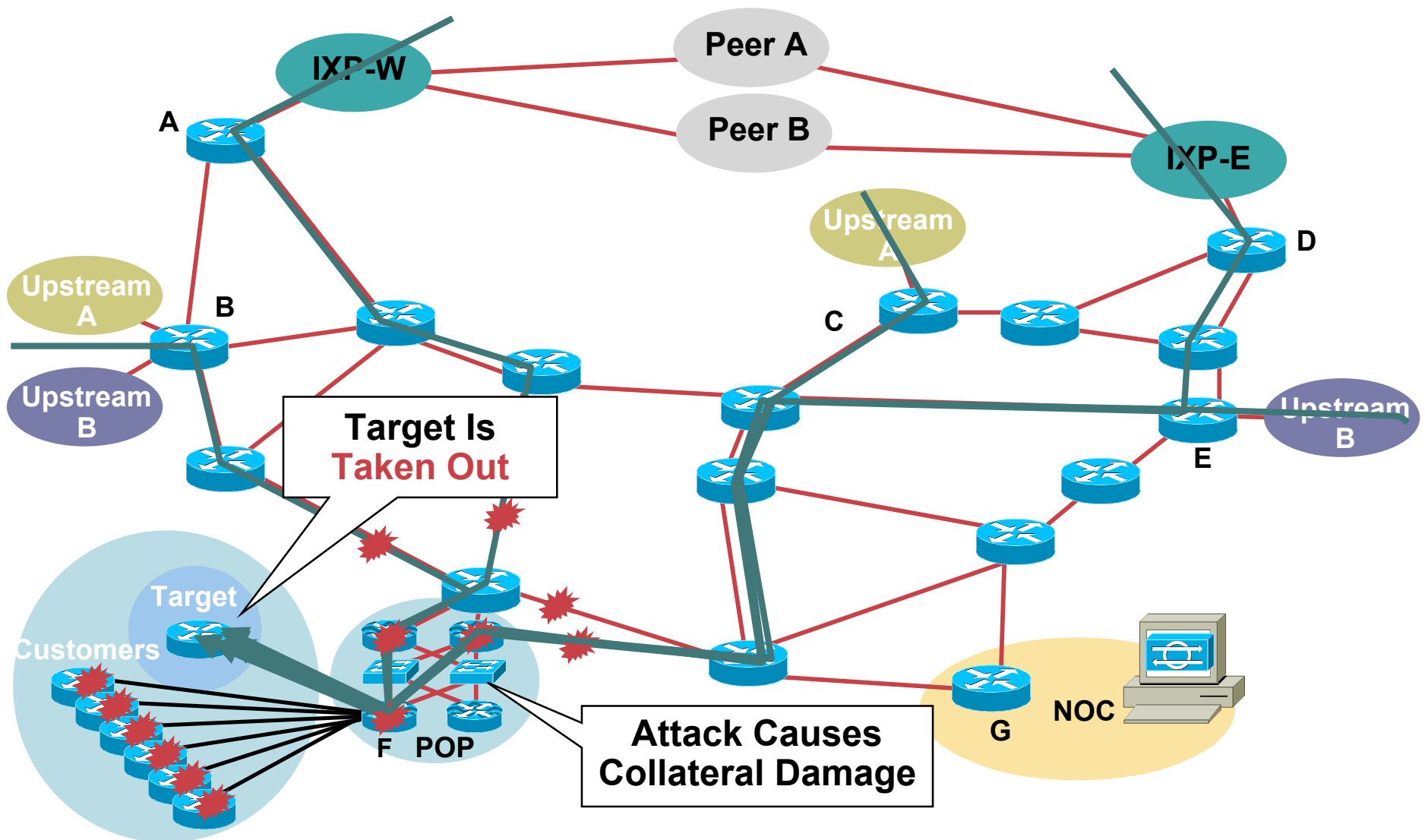
Blackhole Filtering—if the **destination** address equals Null0, we drop the packet

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

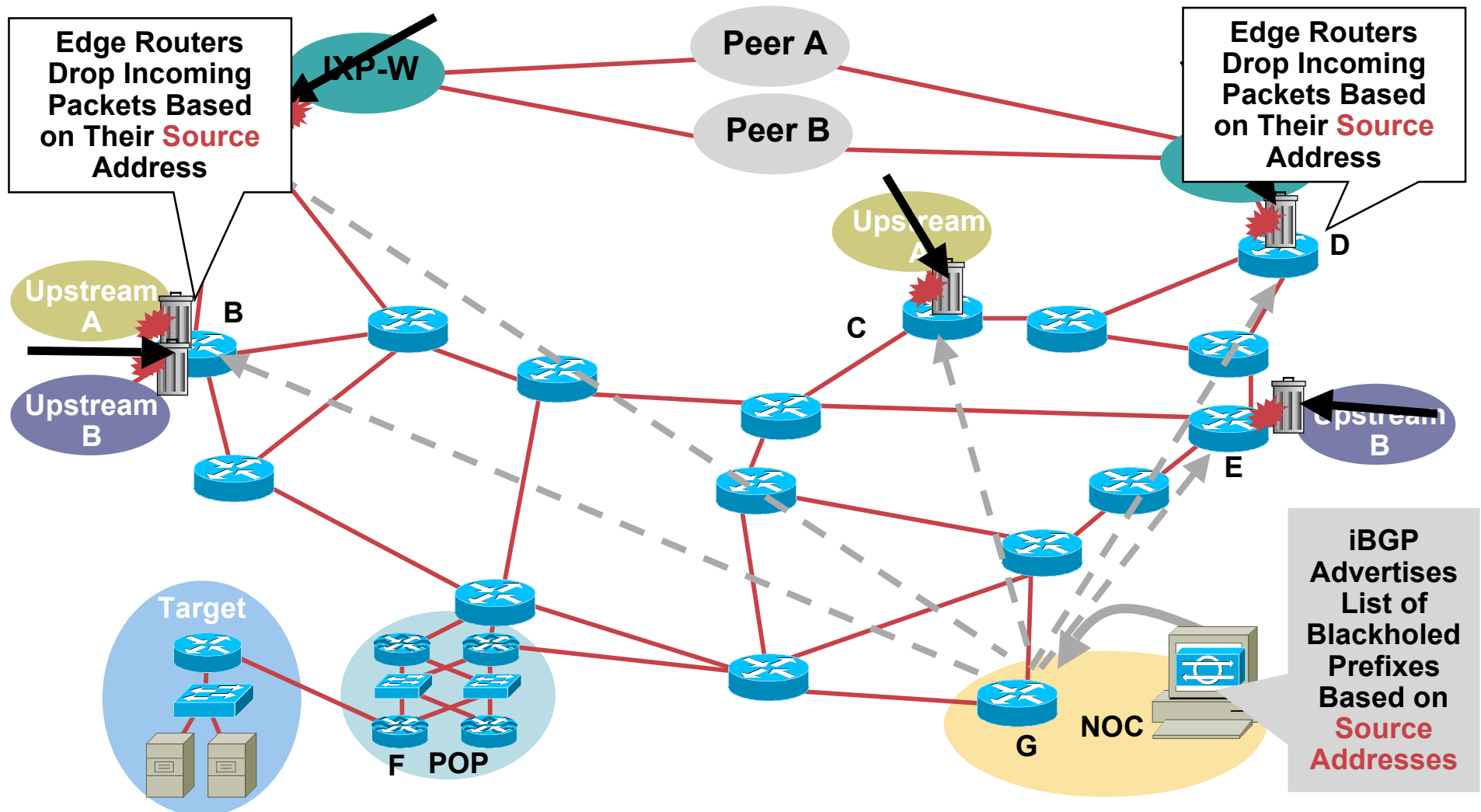
uRPF Loose Check—if the **source** address equals Null0, we drop the packet

- 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



Customer Is DoSed: After Packet Drops Pushed to the Edge



Internal Source-Based Drops

- **Both source and destination drops can be used internally**

Source is likely the most interesting case

Don't forget the Internet/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

Possible Remote Trigger Placement

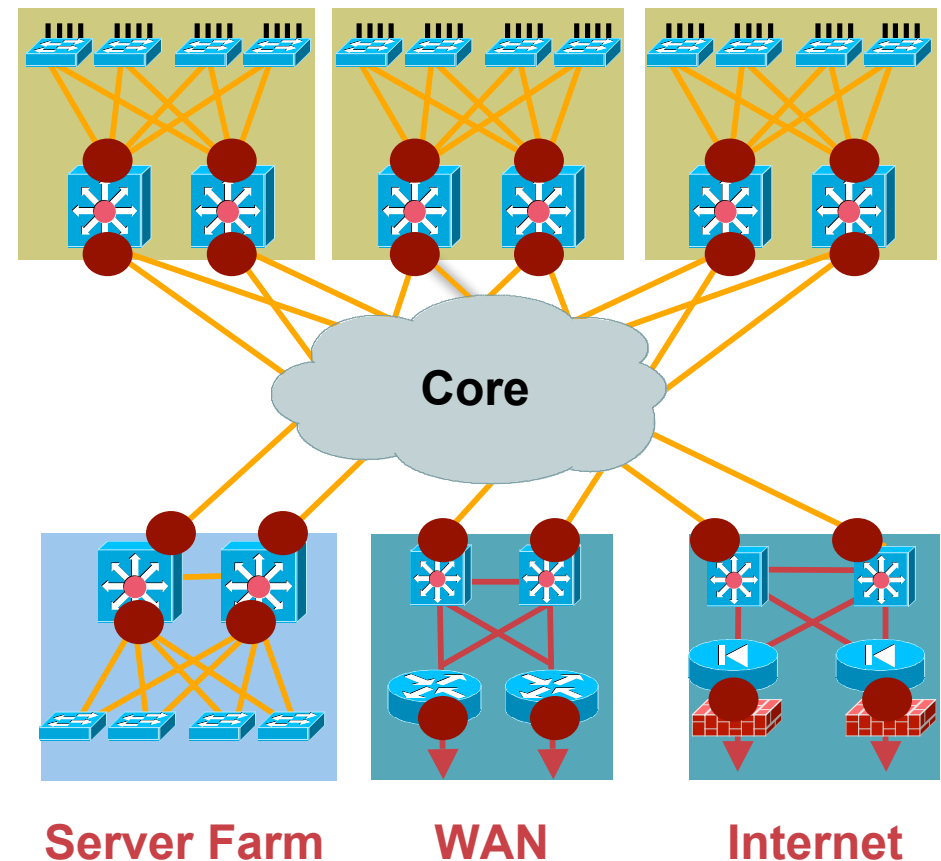
- Dark red dots indicate possible remote drop location
- L3 boundaries between network components

Drop infections at distribution layer

Drop incoming Internet attack at Internet edge

React to incoming attacks from remote office across the WAN

Etc.



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

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

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)**

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**

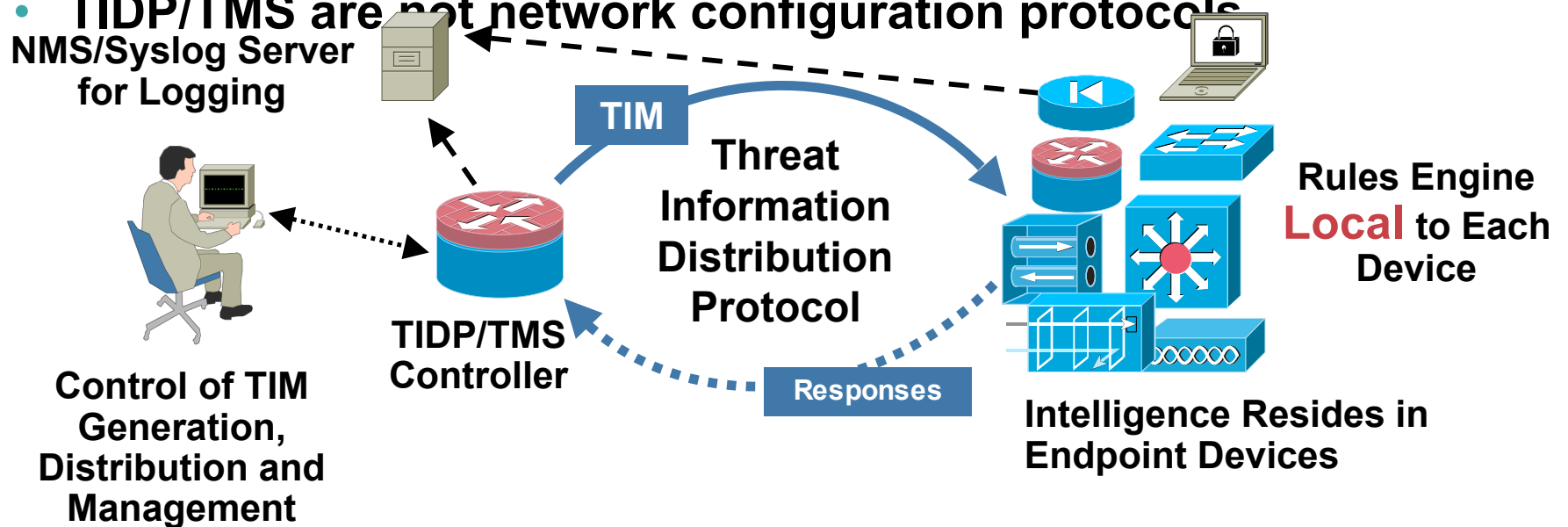
Threat Information Distribution Protocol (TIDP)

- Framework to distribute threat information to network devices
- Distributed from TIDP Mitigation Service (TMS) controller
- Messages authenticated, encrypted, and have replay protection
- Uses TCP port 7548
- Receiving devices configured with unique rule sets
- Uses Threat Information Message (TIM) to ID suspect traffic
- TIM created in threat definition file using XML
- Associates enforcement actions (Block or Redirect) with suspect traffic
- Available in 12.4(6)T

http://www.cisco.com/en/US/products/ps6441/products_feature_guid_e09186a00805ec975.html

TIDP Architecture

- TIDP/TMS distributes device independent Threat Information TIMs through networks
- Each devices uses local device rules to convert TIMs into dynamic device specific enforcement actions
- TIDP/TMS are not network configuration protocols



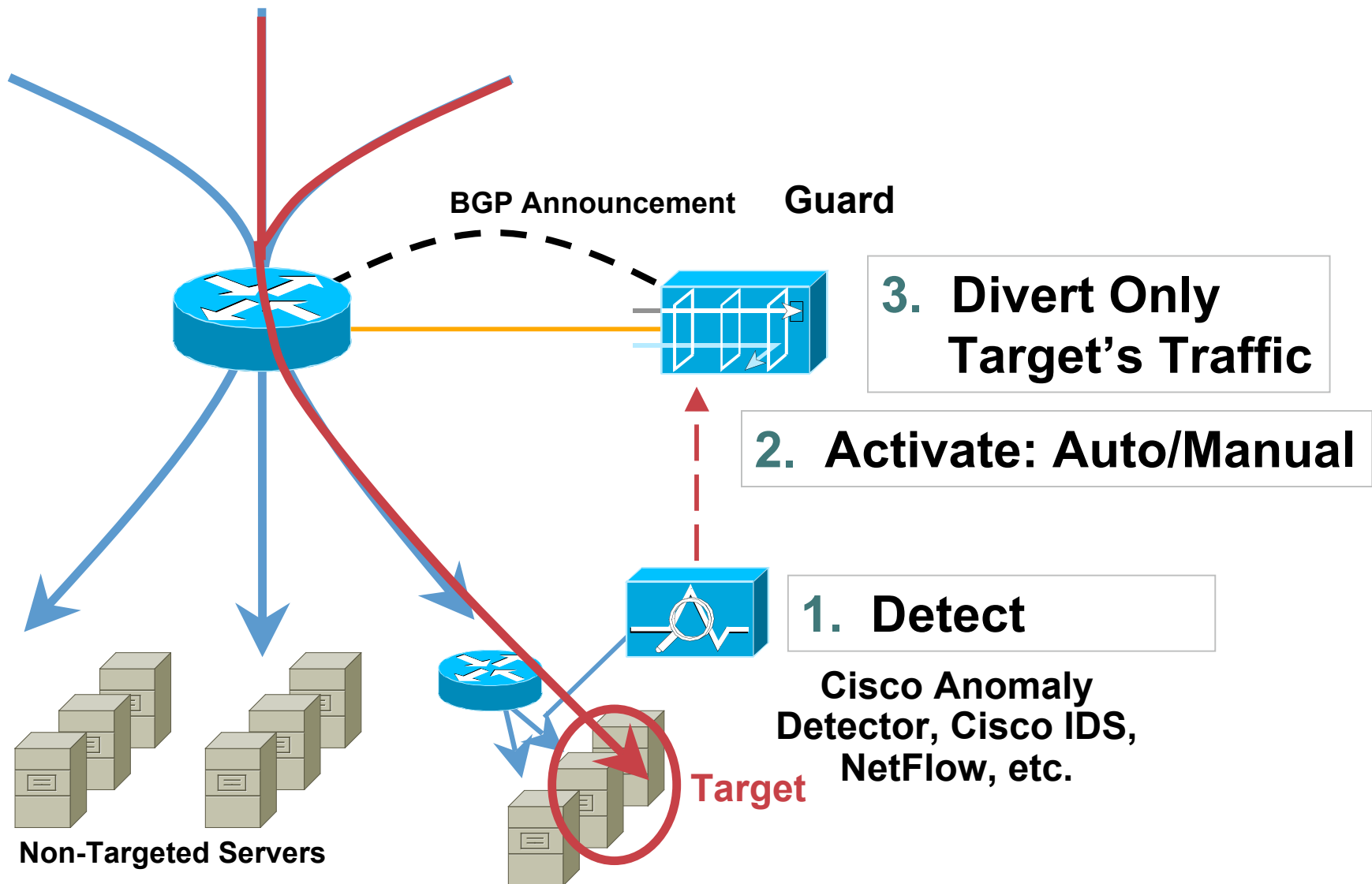
Packet Scrubbing



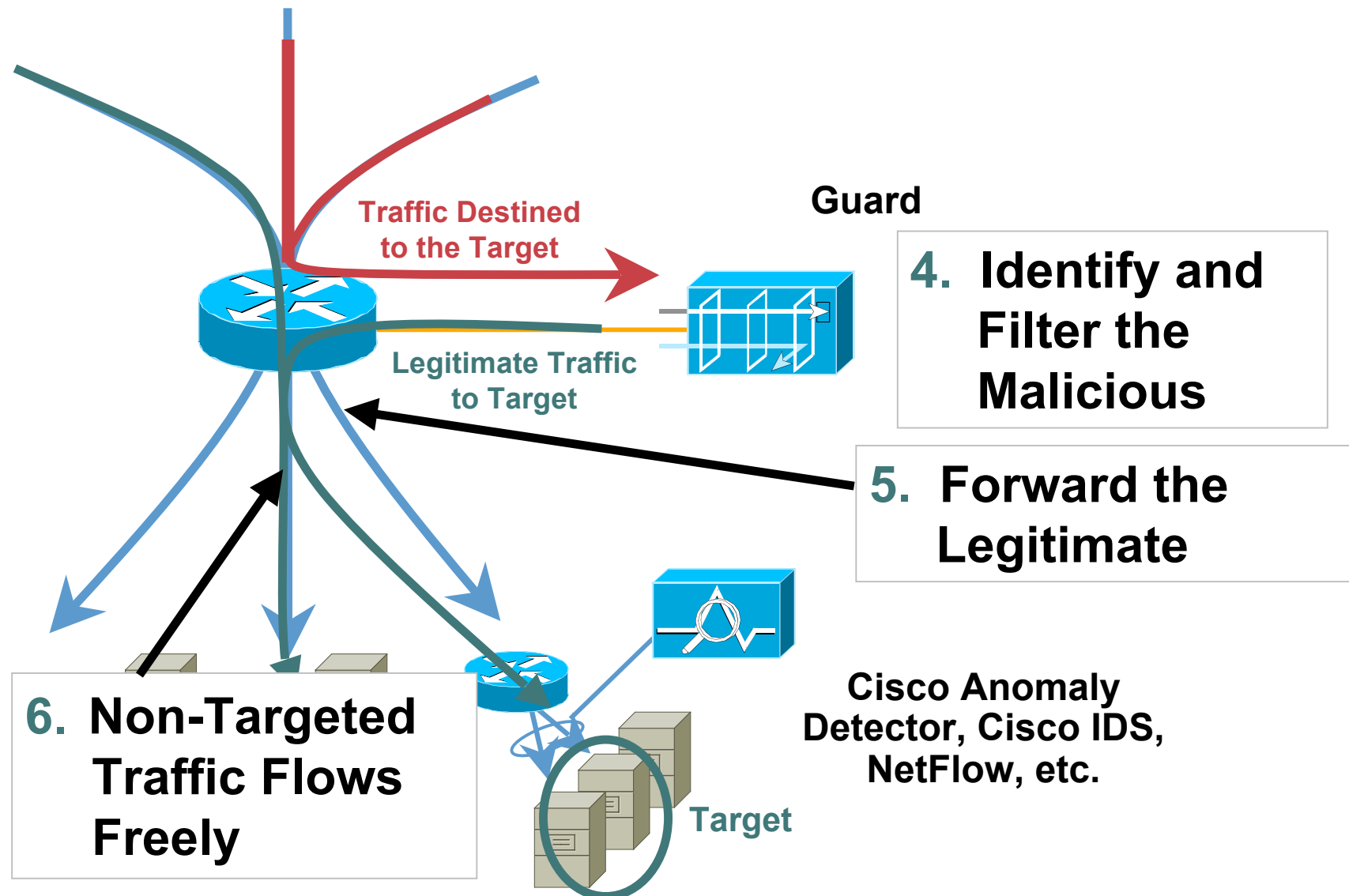
Mitigation: Packet “Scrubbing”

- **Get the packets to the scrubber(s)**
- **Can use the same BGP mechanism to redirect traffic to scrubbing devices**
- **Activate redirection:**
 - Redistribute host route for victim into BGP with next-hop set to scrubbing devices**
 - Route is propagated using BGP to all BGP speaker and traffic redirected**
- **When attack is over, BGP route can be removed to return to normal operation**

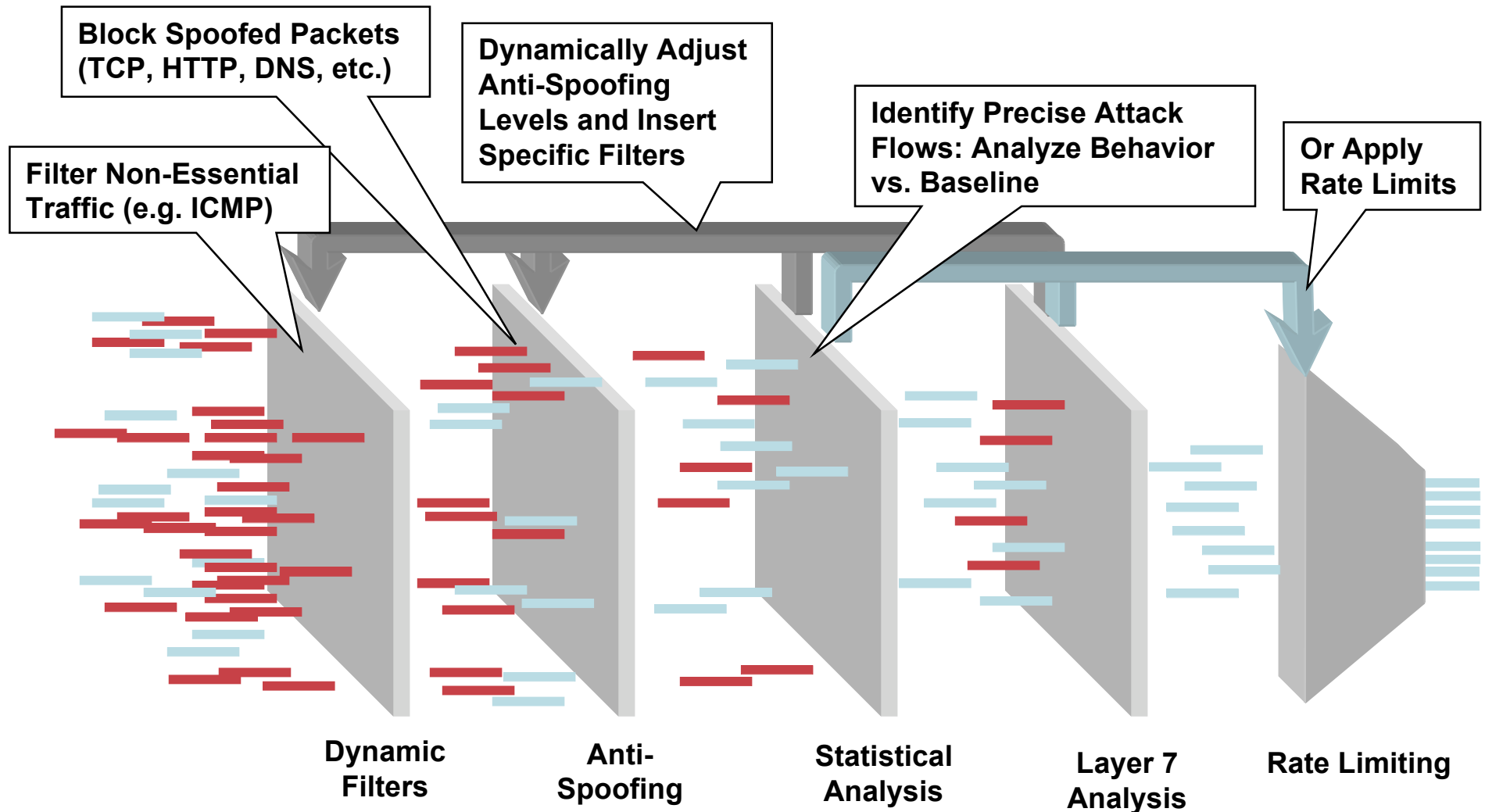
Cisco DDoS Solution—Packet Scrubbing



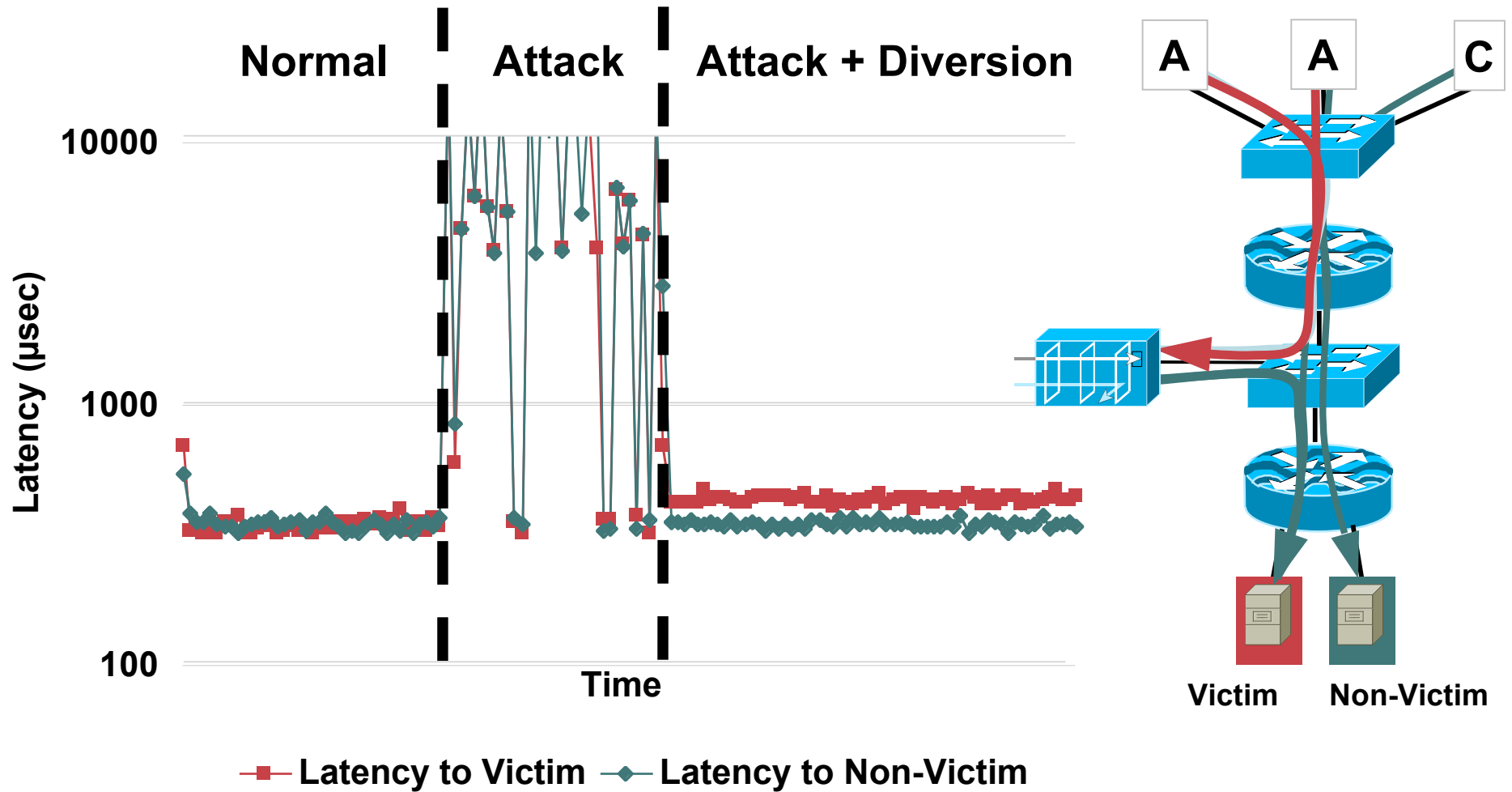
Cisco DDoS Solution—Packet Scrubbing



Cisco DDoS Solution—Multiple Layers of Defense

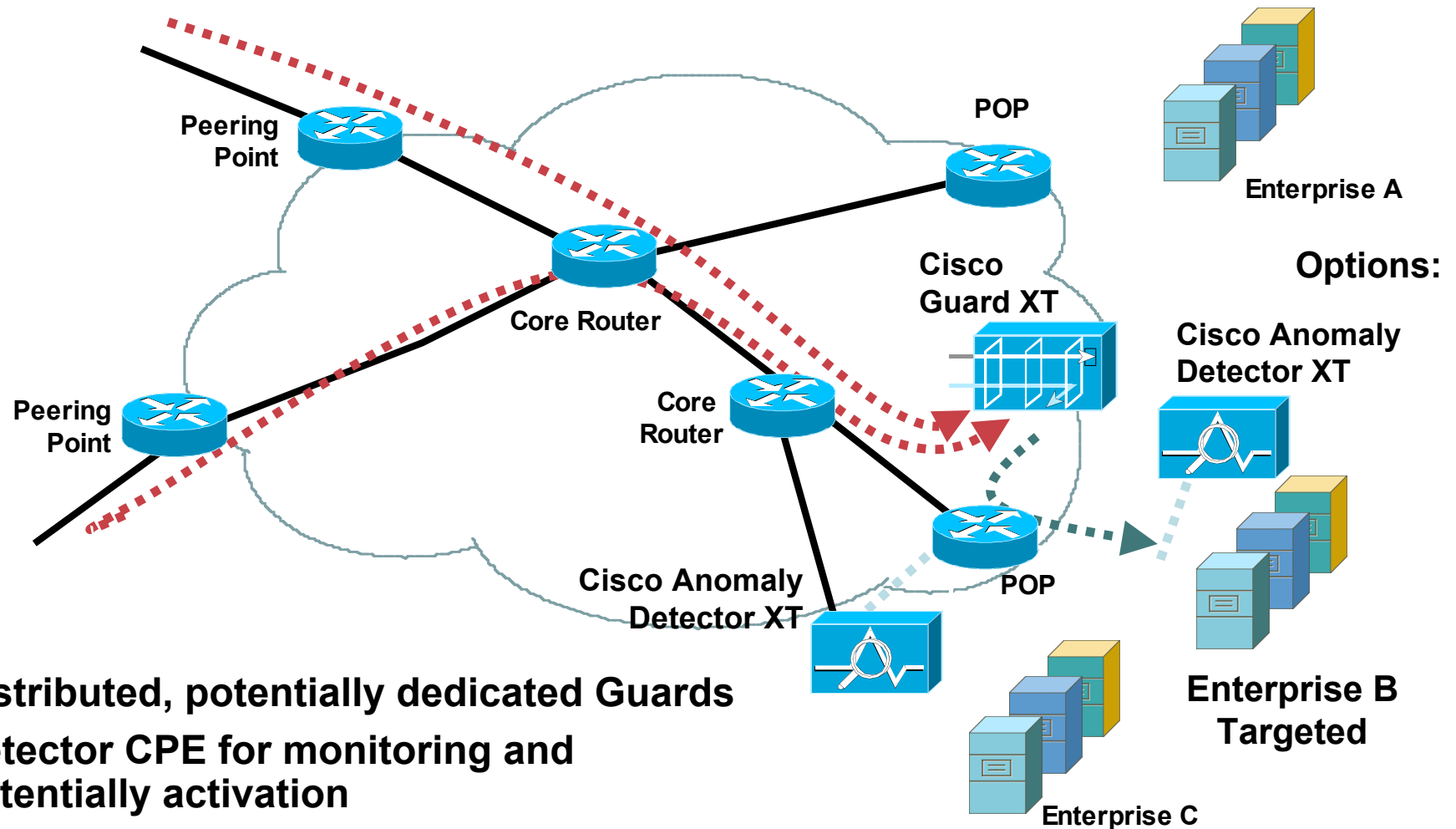


Blackhole Shunt with Scrubber



DDoS Protection

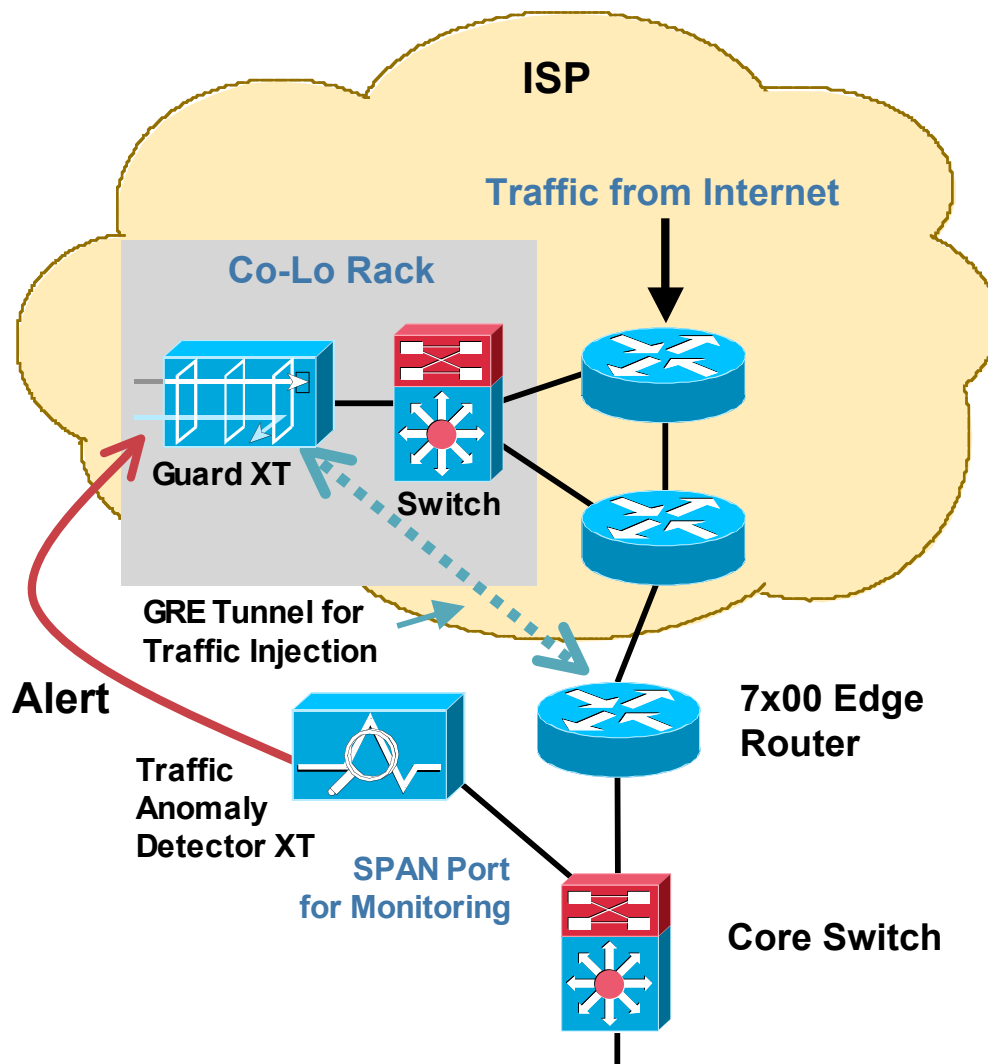
Service Provider Distributed/Edge Protection



- Distributed, potentially dedicated Guards
- Detector CPE for monitoring and potentially activation
- Potentially Detector at SP for monitoring, or NetFlow

DDoS Protection Via Provider Edge Co-Location

- Enterprise controlled, but upstream mitigation protects link and enterprise edge router
- Enterprise located detector activates the guard via separate management circuit
- Additional router isolates routing updates
- GRE Tunnel is configured from guard to enterprise edge router for traffic injection



DDoS Threat Types That Can Be Mitigated by Guard

Bandwidth Consumption Attacks

1. Spoofed and Non-Spoofed Flood Attacks

TCP Flag (SYN, SYN-ACK, ACK, FIN)

ICMP

UDP

Examples: SYN Flood, Smurf, LAND, UDP Flood

2. Zombie/Botnet Attacks

Each zombie or bot source opens multiple TCP connections

Each zombie or bot source opens multiple TCP sessions and issue repetitive HTTP requests

3. DNS Attacks

DNS Request Flood

Resource Starvation Attacks

1. Packet Size Attacks

Fragmented Packets

Large Packets

Examples: Teardrop, Ping-of-Death

2. Low Rate Zombie/Botnet Attacks

Similar to Bandwidth consumption attacks except that each attack source sends multiple requests at low rate

3. DNS Attacks

DNS Recursive Lookup

Reacting to Attacks

- **Many, many reaction mechanisms**
- **Today, we focused on core security-centric techniques**
- **No one tool or technique is applicable in all circumstances**

Think “toolkit”

- **Choose your techniques wisely**
If you only have a hammer, all problems begin to look like nails

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/cgi-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>

- **Cisco SPAN/RSPAN for 6500/7600 documents**

http://www.cisco.com/en/US/products/hw/switches/ps708/products_configuration_guide_chapter09186a008007f323.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/masse_y_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

Security reference information: various white papers on DoS attacks and how to defeat them

<http://www.cisco.com/warp/public/707/ref.html>

- **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