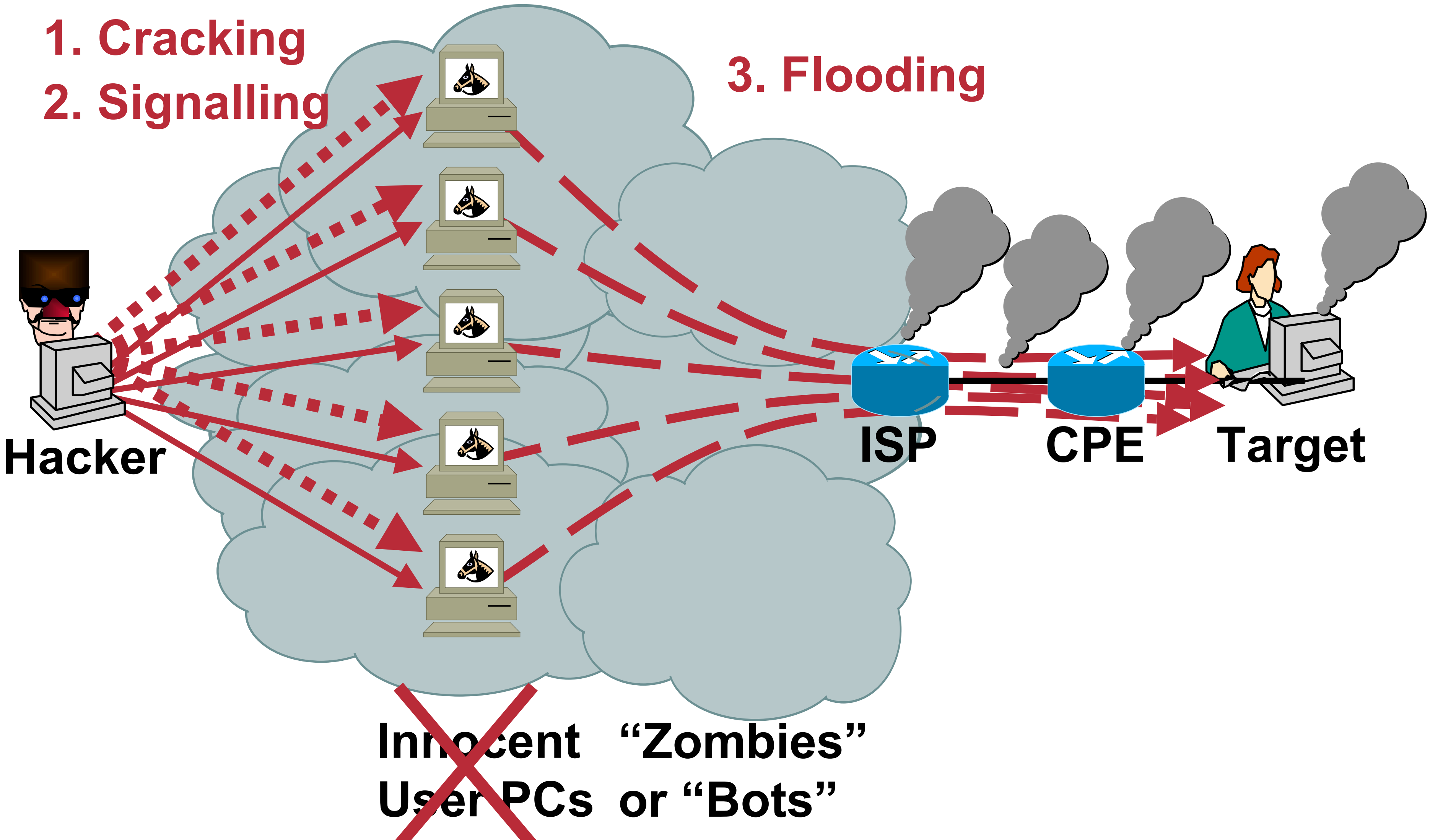
NETWORKERS

CISCO SYSTEMS

Surviving a DoS Attack

SEC-301

DoS: The Procedure



Agenda

- Detecting and Classifying DoS Attacks
- Tracing DoS Attacks
- Containing DoS Attacks
- Special Case: Web Server Protection
- Additional Solutions: Arbor and Riverhead

Disclaimer:
- DoS is a research topic!
- Please contribute your experience!

Detecting and Classifying DoS Attacks

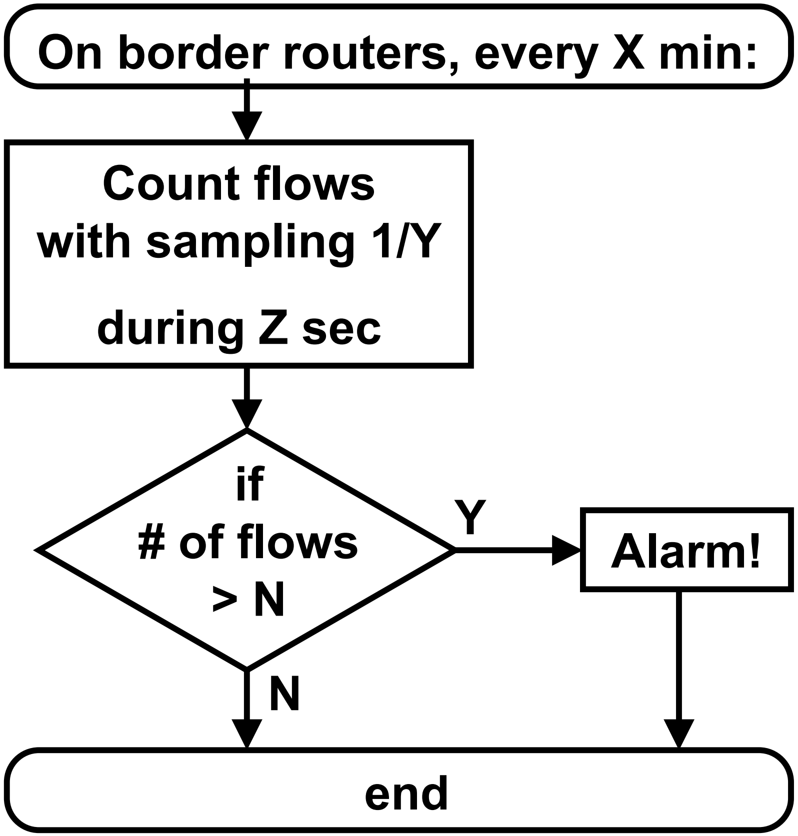
Ways to Detect and Classify DoS Attacks

- Customer Call
- SNMP: Line/CPU overload, Drops
- NetFlow: Counting Flows
- ACLs with Logging
- Backscatter
- Sniffers

Detecting DoS Attacks with NetFlow

Cisco.com

- **Basis: Have NetFlow running on the network**



DANTE uses:
X=15 min, Y=200,
Z=10 sec, N=10

Values are empirical

How does a DoS Attack Look Like?

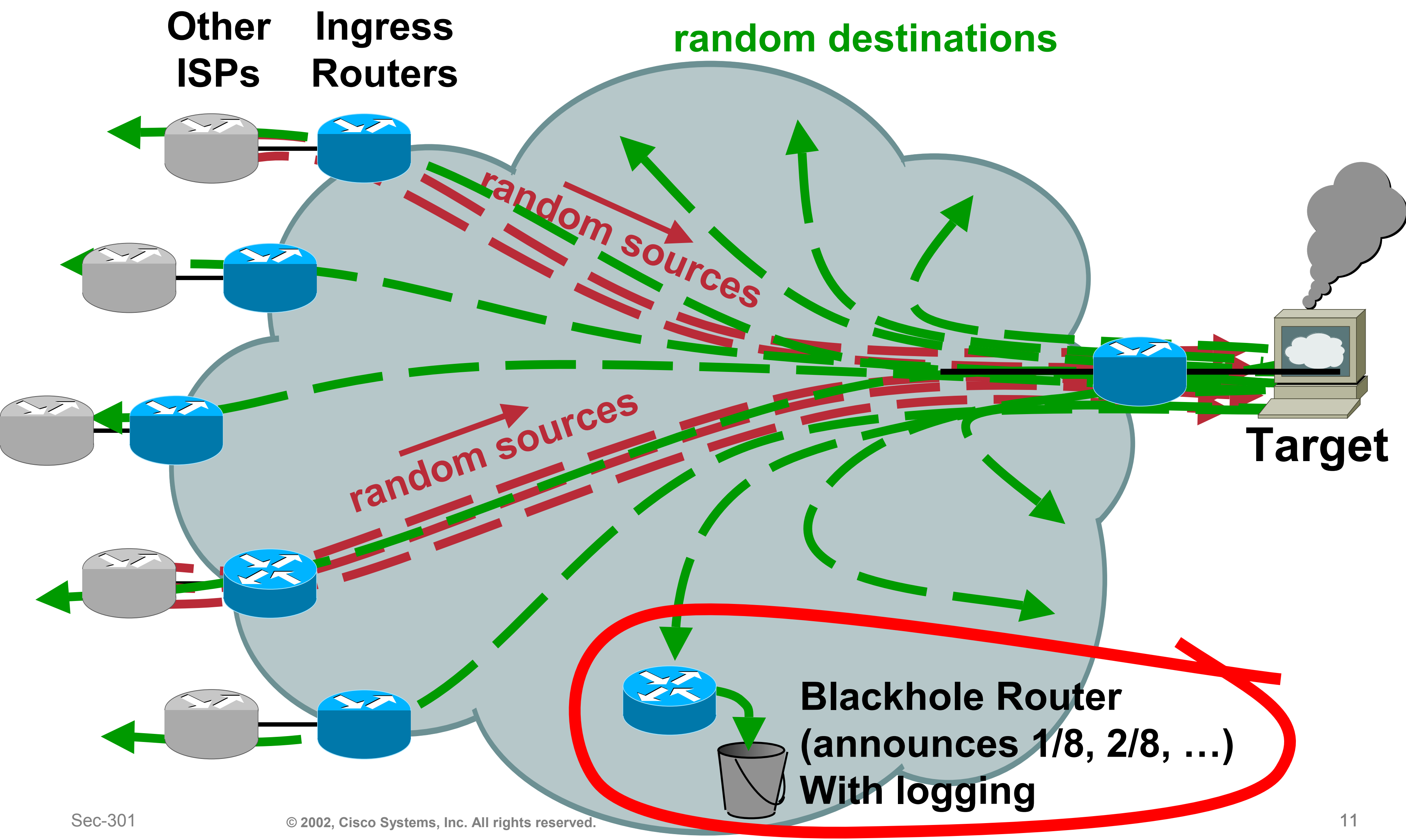
Potential DoS attack (33 flows) on router1
Estimated: 660 pkt/s 0.2112 Mbps
ASxxx is: …
ASddd is: …

src_ip	dst_ip	in int	out int	src port	dest port	pkts	bytes	prot	src_as	dst_as
192.xx.xxx.69	194.yyy.yyy.2	29	49	1308	77	1	40	6	xxx	ddd
192.xx.xxx.222	194.yyy.yyy.2	29	49	1774	1243	1	40	6	xxx	ddd
192.xx.xxx.108	194.yyy.yyy.2	29	49	1869	1076	1	40	6	xxx	ddd
192.xx.xxx.159	194.yyy.yyy.2	29	49	1050	903	1	40	6	xxx	ddd
192.xx.xxx.54	194.yyy.yyy.2	29	49	2018	730	1	40	6	xxx	ddd
192.xx.xxx.136	194.yyy.yyy.2	29	49	1821	559	1	40	6	xxx	ddd
192.xx.xxx.216	194.yyy.yyy.2	29	49	1516	383	1	40	6	xxx	ddd
192.xx.xxx.111	194.yyy.yyy.2	29	49	1894	45	1	40	6	xxx	ddd
192.xx.xxx.29	194.yyy.yyy.2	29	49	1600	1209	1	40	6	xxx	ddd
192.xx.xxx.24	194.yyy.yyy.2	29	49	1120	1034	1	40	6	xxx	ddd
192.xx.xxx.39	194.yyy.yyy.2	29	49	1459	868	1	40	6	xxx	ddd
192.xx.xxx.249	194.yyy.yyy.2	29	49	1967	692	1	40	6	xxx	ddd
192.xx.xxx.57	194.yyy.yyy.2	29	49	1044	521	1	40	6	xxx	ddd
…	…	…	…	…	…	…	…	…	…	…

Backscatter Analysis

- **Blackhole router:**
  Statically announce *unused* address space (1/8, 2/8, 5/8, …)
  (see http://www.iana.org/assignments/ipv4-address-space)
- **Note: Hackers know this trick: Use also unused space from your own ranges!!!**
- **Victim replies to random destinations**
- **-> Some backscatter goes to blackhole router, where it can be analysed**

Backscatter Analysis



Case Study: Slapper Worm (Sep 2002)

Cisco.com

IP Address: a.b.0.0

```
int isreal(unsigned long server) {
  . . .
      if (a == 127 || a == 10 || a == 0)
        return 0;
      if (a == 172 && b >= 16 && b <= 31)
        return 0;
      if (a == 192 && b == 168) return 0;
      return 1;
}

  . . .
if (!isreal(udpclient.in.sin_addr.s_addr)) break;
 . . .
```

Worm does not use:

127.x.x.x

10.x.x.x

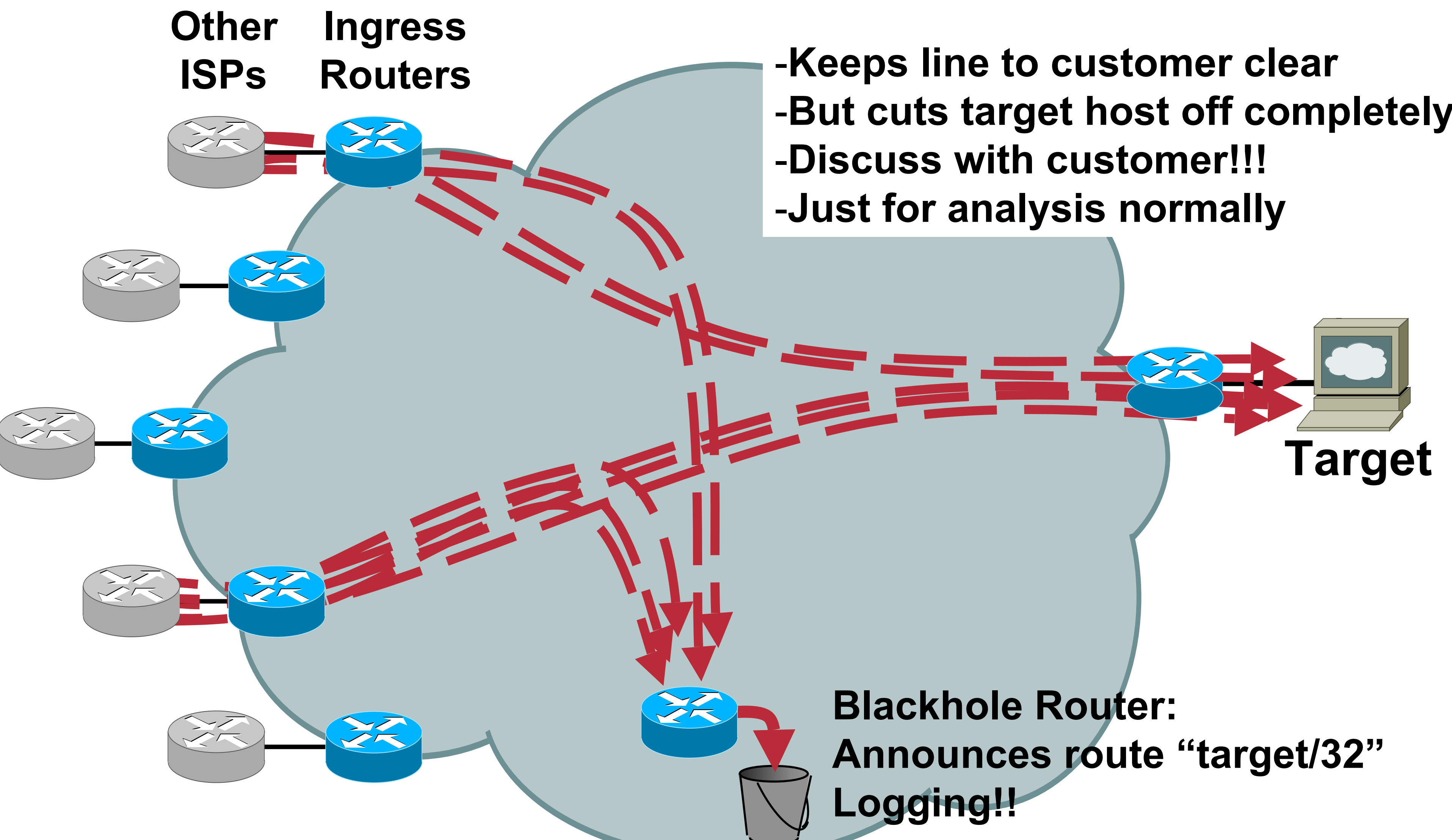
0.x.x.x

172.16-31.x.x

192.168.x.x

Source: http://isc.incidents.org/analysis.html?id=167

Re-Directing Traffic from the Victim



Tracing DoS Attacks

Tracing DoS Attacks

- **If source prefix is not spoofed:**

  -> Routing table
  -> Internet Routing Registry (IRR)
  -> direct site contact

- **If source prefix is spoofed:**

  -> Trace packet flow through the network
  ACL, NetFlow, IP source tracker

  -> Find upstream ISP
  -> Upstream needs to continue tracing

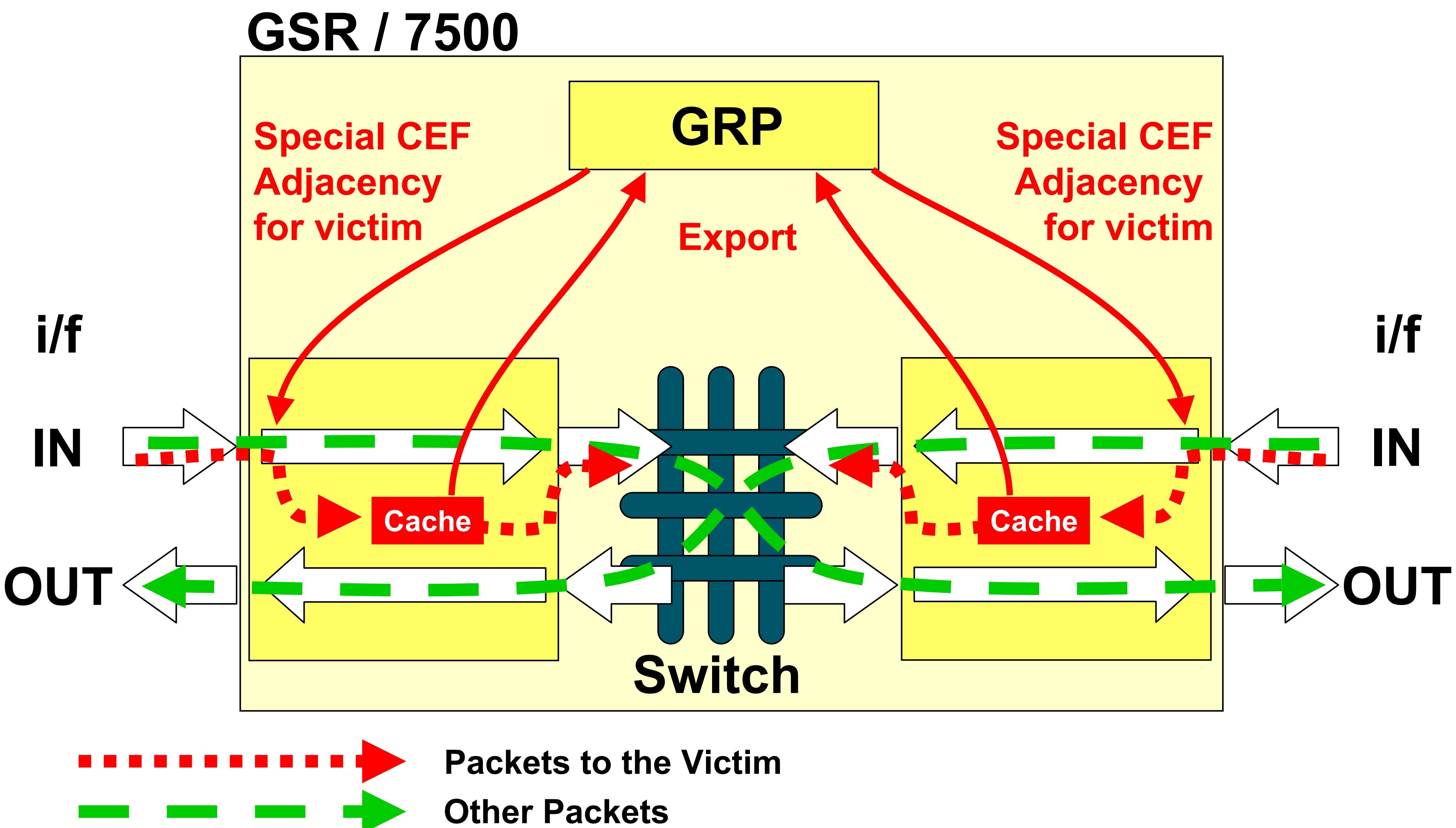
IP Source Tracker

- **Traditional way of tracking DoS: ACL or NetFlow**
  - **Limitation in performance and cross LC support**

Line Card

- **Source Tracker:**
  - **Across LCs, low performance impact**
- **Availability:**
  - **GSR E0,1,2,4: From 12.0(21)S**
  - **GSR E3: From 12.0(24)S (slipped from (23))**
  - **GSR E4+: From 12.0(21)S (POS), (23)S (other)**
  - **Other platforms to follow**

IP Source Tracker (E3: (24)S, E4+ non-POS: (23)S, rest+7500: (21)S)



IP Source Tracker: Config

Enable the feature

Router# ip source-track <victim>

Router# show ip source-track 10.1.2.1 (also: … summary)

Address	SrcIF	Bytes	Pkts	Bytes/s	Pkts/s
10.1.2.1	Pos 1/0	2000	20	200	1

Protocol	Total Flows	Flows /Sec	Packets /Flow	Bytes /Pkt	Packets /Sec	Active(Sec) /Flow	I /I

ICMP	100	1	10	100	10	0	5

Victim

Ingress interface

See: http://www/univercd/cc/td/doc/product/software/ios120/120newft/120limit/120s/120s21/ipst.htm

Tracing Back with NetFlow

- Routers need NetFlow enabled

Victim

router1#sh ip cache flow | include <destination>

Se1 <source> Et0 <destination> 11 0013 0007 159

…. (lots more flows to the same destination)

The flows come from serial 1

Find the upstream router on serial 1

router1#sh ip cef se1

Prefix	Next Hop	Interface
0.0.0.0/0	10.10.10.2	Serial1
10.10.10.0/30	attached	Serial1

Continue on this router

Show ip cache flow

Cisco.com

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

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

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

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

Source Interface

Flow info summary

Flow details

Show ip cache verbose flow

Cisco.com

```
router_A#sh ip cache verbose flow
IP packet size distribution (23597 total packets):
   1-32   64   96  128  160  192  224  256  288  320  352  384  416  448  480
   .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000

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

IP Flow Switching Cache, 278544 bytes
  1323 active, 2773 inactive, 23533 added
  151644 ager polls, 0 flow alloc failures
  Active flows timeout in 30 minutes
  Inactive flows timeout in 15 seconds
  last clearing of statistics never
Protocol         Total    Flows   Packets Bytes  Packets Active(Sec) Idle(Sec)
--------         Flows     /Sec     /Flow  /Pkt     /Sec     /Flow     /Flow
TCP-other        22210      3.1         1  1440      3.1       0.0      12.9
Total:           22210      3.1         1  1440      3.1       0.0      12.9
```

Port Msk AS Port Msk AS NextHop B/Pk Active

```
5FA7 /0  0                 0007 /0  0      0.0.0.0               1440     0.0
Et0/0          175.182.253.65  Se0/0          192.168.1.1    06 00  10         1
```

Tracing Back with ACLs

- **Create ACL:**
  access-list 101 permit ip any <target> log-input

- **Apply to interface for a few seconds:**
  interface xxx
  ip access-group 101 in
  (wait a few seconds)
  no ip access-group 101

- **Log shows interface the attack comes from**

```
14:17:21: %SEC-6-IPACCESSLOGP: list 101 permitted tcp 105.12.73.84(0) (FastEthernet0/0
    0006.d780.2380) -> 192.168.1.1(0), 1 packet
14:17:22: %SEC-6-IPACCESSLOGP: list 101 permitted tcp 166.159.237.65(0) (FastEthernet0/0
    0006.d780.2380) -> 192.168.1.1(0), 1 packet
```

MAC address

src interface

Tracing Back with ACLs (again, bigger)

…

14:17:21: %SEC-6-IPACCESSLOGP: list 101 permitted tcp 105.12.73.84(0) (FastEthernet0/0 0006.d780.2380) -> 192.168.1.1(0), 1 packet

14:17:22: %SEC-6-IPACCESSLOGP: list 101 permitted tcp 166.159.237.65(0) (FastEthernet0/0 0006.d780.2380) -> 192.168.1.1(0), 1 packet

…

source interface

MAC address of upstream router

Note: ACL with "log" does not show the source interface, "log-input" does (see above)

Tracing Back Across an IXP
(…or any other shared medium)

- **NetFlow: Shows i/f only**

  Useless if IXP: Lots of routers behind…

- **ACLs with log-input:**

  Shows also the MAC address of the router:

1d00h: %SEC-6-IPACCESSLOGDP: list 101 denied icmp 11.1.1.18 (Ethernet0 0001.96e6.7641) -> 10.1.2.1 (0/0), 169 packets

GSR6#sh arp | include 0001.96e6.7641
Internet 12.1.1.99 152 0001.96e6.7641 ARPA Ethernet0

Originating router

Trace-Back in One Step: ICMP Backscatter

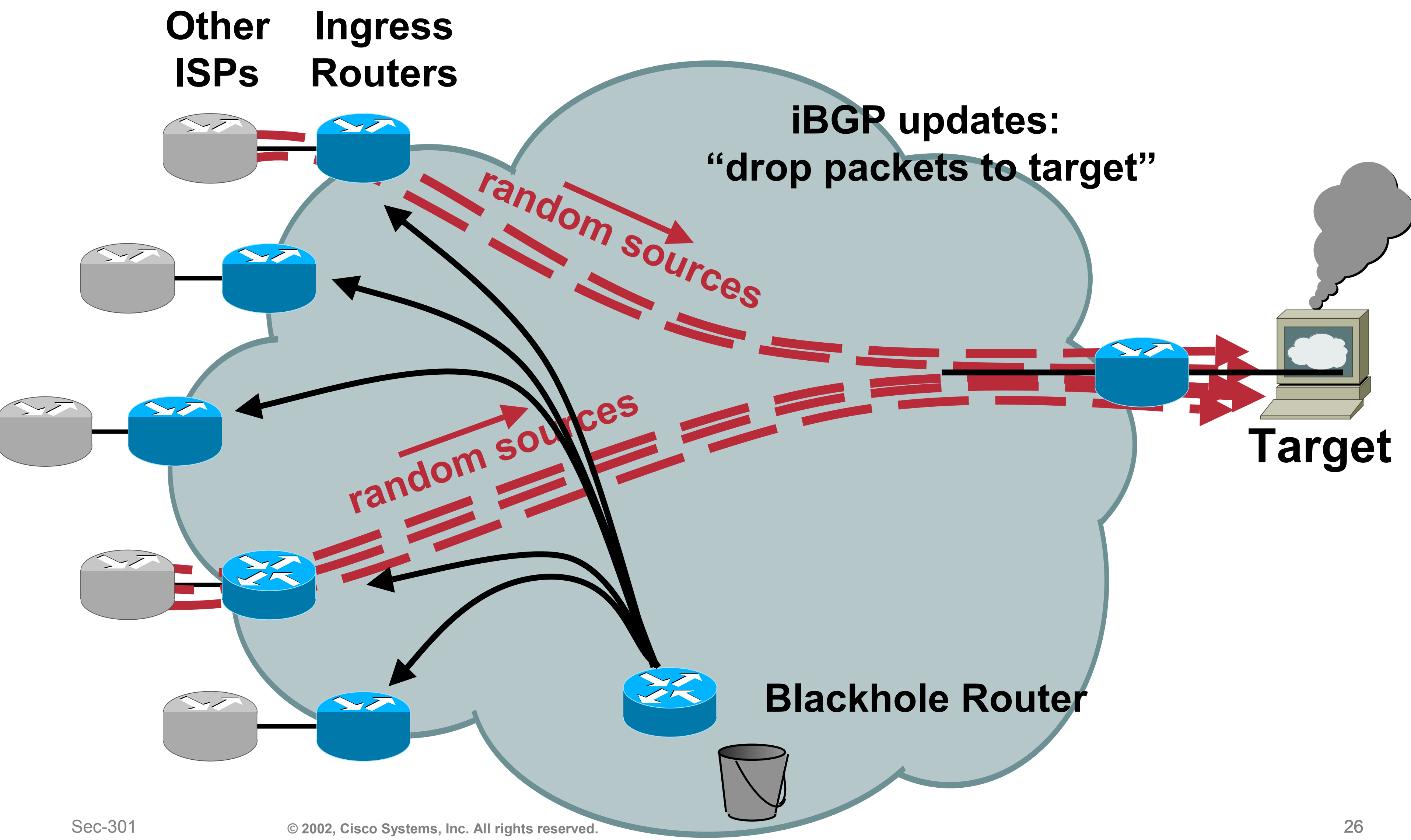
Cisco.com

- Border routers: Allow ICMP (rate limited)
- From Black hole router:

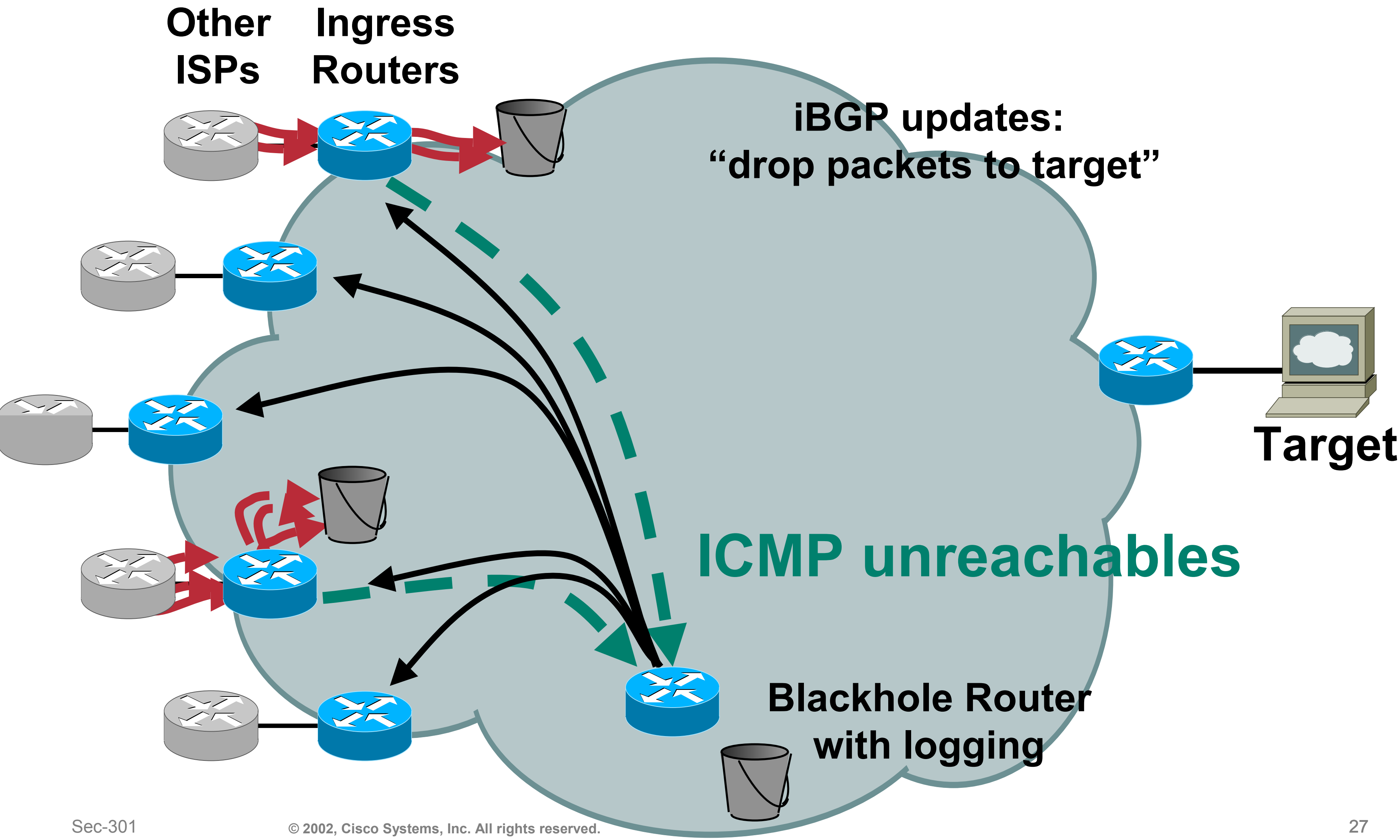
  iBGP update to all ingress routers:
  “drop all traffic to <victim>” (details later)

- All ingress router drop traffic to <victim>
- And send ICMP unreachables to source!!
- Black hole router logs the ICMPs!

Trace-Back in One Step: ICMP Backscatter



Trace-Back in One Step: ICMP Backscatter



Trace-Back in One Step: ICMP Backscatter

Cisco.com

On black hole router:

- Static routes for 1/8,2/8,5/8 (will attract 3/256 of packets)
- access-list 105 permit icmp any any log-input
  access-list 105 permit ip any any

- Border router sends ICMP unreachable for dropped packets, to source.
- If source is random, some will go to 1/8, 2/8, 5/8, …

03:17:22: %SEC-6-IPACCESSLOGDP: list 105 permitted icmp 192.168.0.2
(Serial0/0 *HDLC*) -> 5.52.203.66 (0/0), 1 packet
03:17:38: %SEC-6-IPACCESSLOGDP: list 105 permitted icmp 192.168.0.2
(Serial0/0 *HDLC*) -> 1.167.111.47 (0/0), 1 packet
03:17:52: %SEC-6-IPACCESSLOGDP: list 105 permitted icmp 192.171.12.5
(Serial0/1 *HDLC*) -> 2.153.59.34 (0/0), 1 packet
…

Summary Tracing DoS Attacks

- **Non-spoofed: Technically trivial (IRR)**

  But: Potentially tracing 100’s of sources…

- **Spoofed:**

  IP Source Tracker

  NetFlow:
  Trivial if mechanisms are installed
  Manually: Router by router

  ACLs:
  Has performance impact on some platforms
  Mostly manual: Router by router

  Backscatter Technique:
  One step, fast

Cisco.com

Containing DoS Attacks

Prevention of Address Spoofing

- **All ISPs should do one of:**

  Unicast Reverse Path Fowarding (uRPF) Check

  Packet filters (ACLs)

on all external i/f (where possible)

-> See ISP Essentials:
http://www.cisco.com/public/cons/isp/documents/

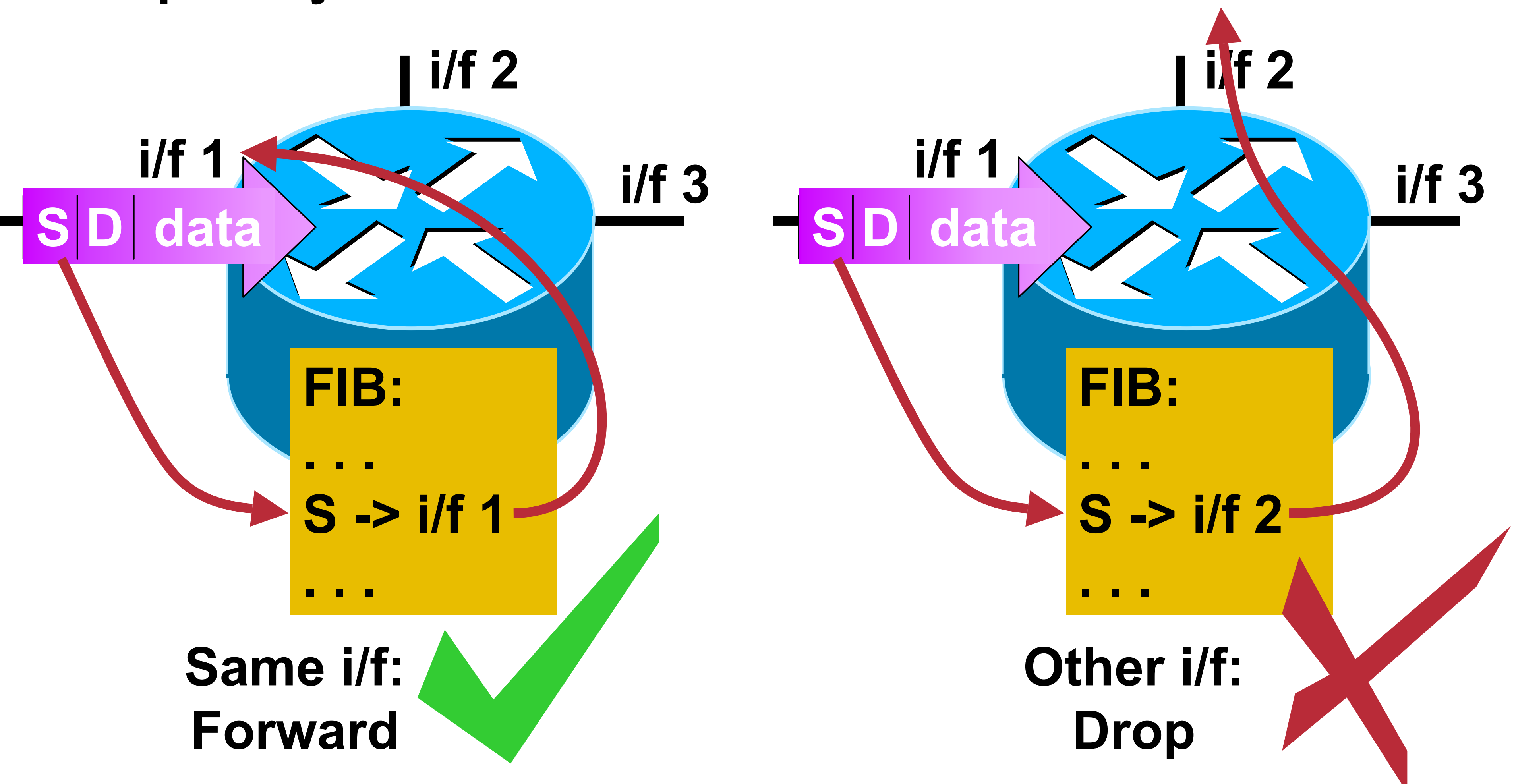
Note: This does NOT prevent DoS!

(just spoofed packets)

Strict uRPF Check (Unicast Reverse Path Forwarding)

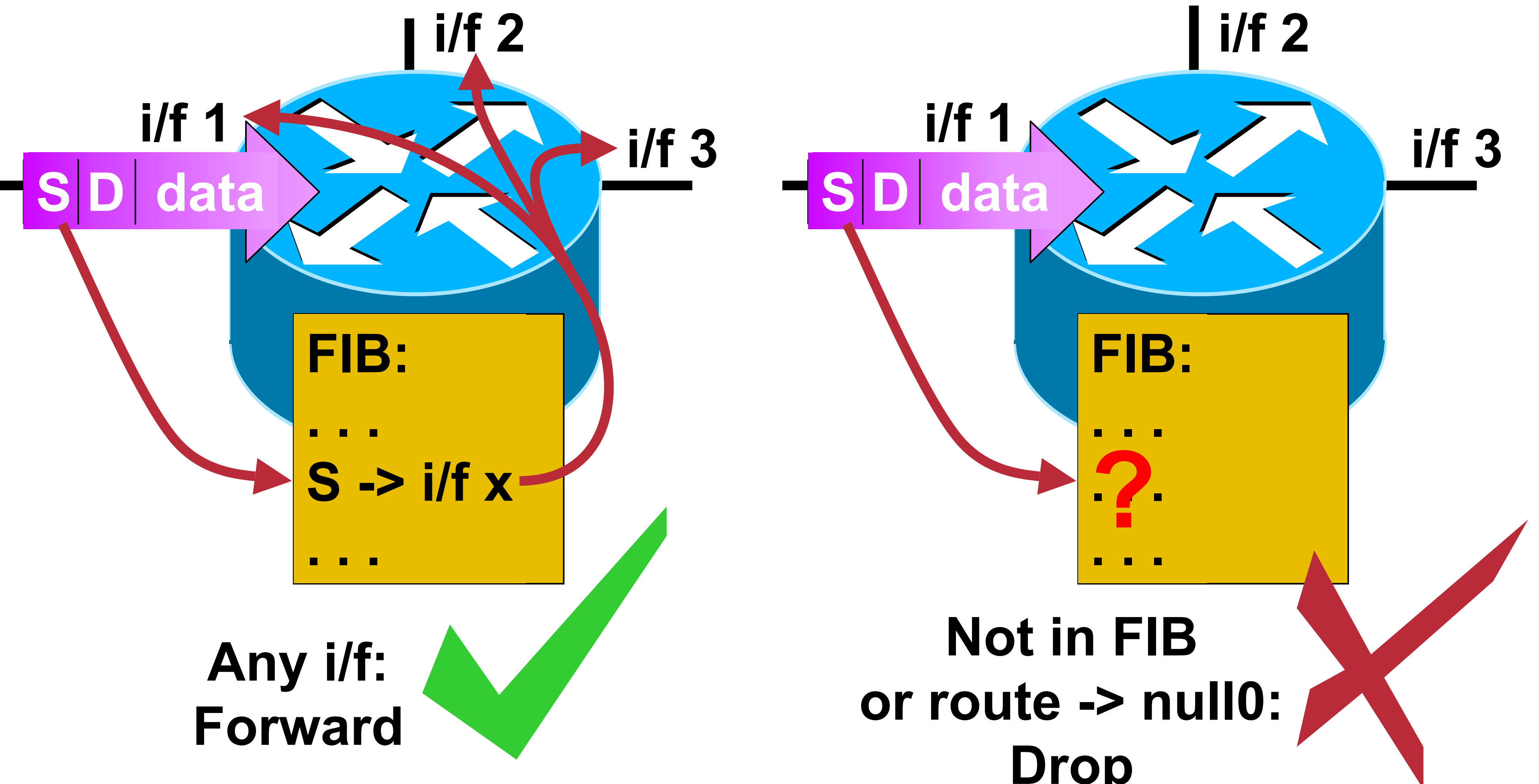
router(config-if)# **ip verify unicast reverse-path**

or: **ip verify unicast source reachable-via rx allow-default**



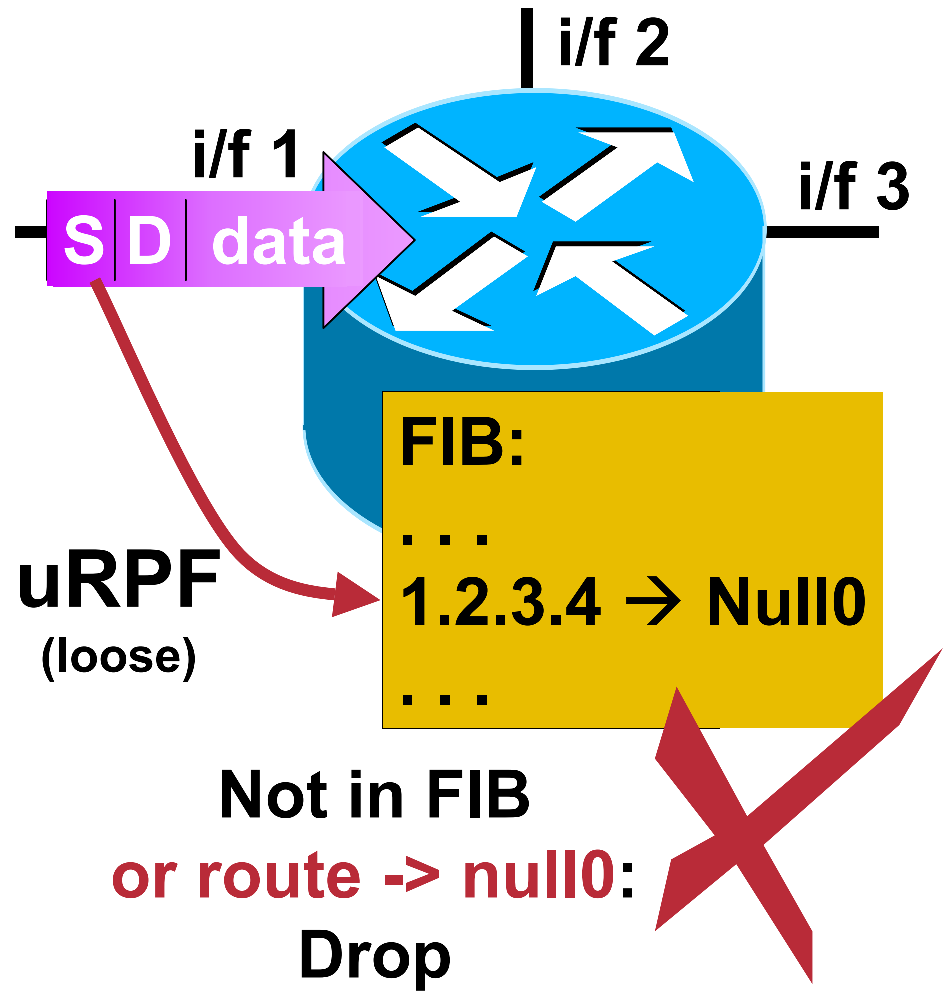
Loose uRPF Check (Unicast Reverse Path Forwarding)

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



Deleting Traffic *from* a Source Address

- Goal: Delete all packets *from* 1.2.3.4
- Static route: 1.2.3.4 → Null0
- Loose uRPF: "reachable-via any"
- Minimal CPU impact (2-3%), CEF based
- Alternative to ACL



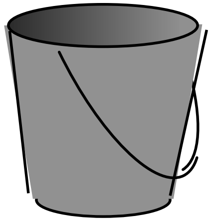
Effectiveness of uRPF

> “ We're currently taking a ~25Mb/s attack, and when I put an ACL entry on the ingress interface the CPU load hit 95%. I switched over to distributing a route to a next-hop that tied the CEF adjacency to Null0, and the traffic was still discarded and CPU utilization went down to 45% (20% is normal for the box, when we're not being attacked).
>
> Turns out I do understand it. It's very cool. :-) ”

A customer

Using CAR to Rate Limit Attack Traffic


```
interface xy
  rate-limit output access-group 2020 3000000
          512000 786000 conform-action transmit
                          exceed-action drop
access-list 2020 permit icmp any any echo-reply
```



- **Other ACLs for other attacks: UDP based attacks, …**
- **Watch your CPU!!!**

Using CAR against SYN Flood Attacks

Cisco.com

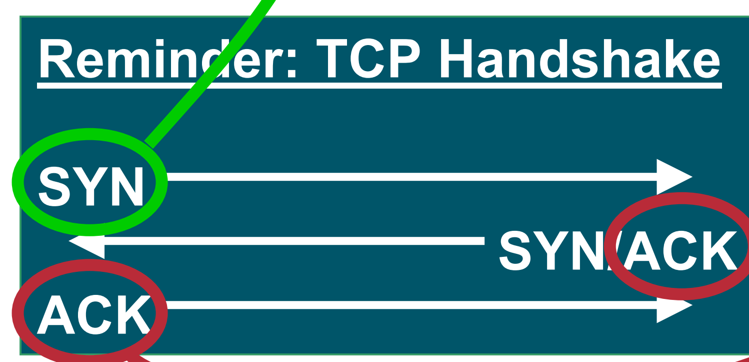
- **Same rate limiting, with this ACL:**

  access-list 169 deny tcp any any established

  access-list 169 permit tcp any host victim-host

  access-list 169 deny ip any any

Watch your CPU!



Stopping SYN Attacks: TCP intercept

- SYN rate-limiting: Kills “good” traffic, too
- Proper solution: TCP intercept
- Performance:

Routers:	x,000 pps	Might not suffice! Check data sheet!
PIX:	x,000 pps	
CSS:	x,000 pps	
CSM:	70,000 pps	Very good!

- CSM scales: Several blades per Cat6k
- DoS against Web? -> Content story!!!

Important!

Making it Scalable

- Problem: Potentially 100's of sources to track and shun
- Manually: For few sources only
- On big ISP networks:
  Scalable mechanisms required!
- Idea: Use routing to distribute information

Configuring CAR through BGP

Cisco.com

- Feature: QPPB (QoS Policy Propagation with BGP)
- On each border router: Define a CAR policy on each border router, linked to a QoS group (normally unused)
- To limit an attacking network, assign this network to the QoS group (BGP community)

See: http://www.cisco.com/univercd/cc/td/doc/product/software/ios111/cc111/bgpprop.htm

Shunning with uRPF and BGP

Cisco.com

- ip route x.x.x.x null0 is manual :-(
- BGP cannot send “next-hop null0”… but:
- BGP can send “next-hop 192.0.2.1”
- And on each border router:

  ip route 192.0.2.1 null0

- Router receives iBGP routing update:

  “Route x.x.x.x next-hop 192.0.2.1” (comm: local-AS)

  and it has an ip route 192.0.2.1 null0

  Thus: x.x.x.x -> null0 (note: CEF required!)

- With uRPF: Source x.x.x.x also -> null0

Trick: not in use!

Effect of BGP Remote Trigger

- Traffic to/from a specific subnet will be sent to null0
- Automatically, on all border routers
- No attack traffic on backbone
- But… *Where is the attack coming from??? Which upstream ISPs to notify???*

ICMP Backscatter

On black hole router:

- Static routes for 1/8,2/8,5/8 (will attract 3/256 of packets)
- access-list 105 permit icmp any any log-input
  access-list 105 permit ip any any

- Border router sends ICMP unreachable for deleted packets, to source.
- If source is random, some will go to 1/8, 2/8, 5/8, …

03:17:22: %SEC-6-IPACCESSLOGDP: list 105 permitted icmp 192.168.0.2 (Serial0/0 *HDLC*) -> 5.52.203.66 (0/0), 1 packet
03:17:38: %SEC-6-IPACCESSLOGDP: list 105 permitted icmp 192.168.0.2 (Serial0/0 *HDLC*) -> 1.167.111.47 (0/0), 1 packet
03:17:52: %SEC-6-IPACCESSLOGDP: list 105 permitted icmp 192.171.12.5 (Serial0/1 *HDLC*) -> 2.153.59.34 (0/0), 1 packet
…

ICMP Rate-Limiting

GSR6(config)# ip icmp rate-limit unreachable 3000

Unreachables sent every 3000 ms

```
Reply from 12.1.1.6: Destination net unreachable.
Request timed out.
Request timed out.
Reply from 12.1.1.6: Destination net unreachable.
Request timed out.
Request timed out.
Reply from 12.1.1.6: Destination net unreachable.
Request timed out.
Request timed out.
Reply from 12.1.1.6: Destination net unreachable.
Request timed out.
Request timed out.
Reply from 12.1.1.6: Destination net unreachable.
Request timed out.
Request timed out.
```

Summary: Containing DoS Attacks

Cisco.com

- **ACLs:**

    Manual, on some routers performance impact

- **uRPF:**

    Stops non-existing sources

    Automated with BGP for specific shunning

- **CAR:**

    Limit attack flow, performance impact

    Manual or automated via QPPB (BGP)

Reality Check: Does all this make sense?

- **Rate Limit:**

  Mostly we also limit “good” traffic

  --> Users are also limited

  Exceptions: ICMP, maybe UDP?

- **ACLs and Null0:**

  We drop all traffic to a server

  Also “good” traffic

  Goal: ACLs as specific as possible

We need to become “smarter”!!

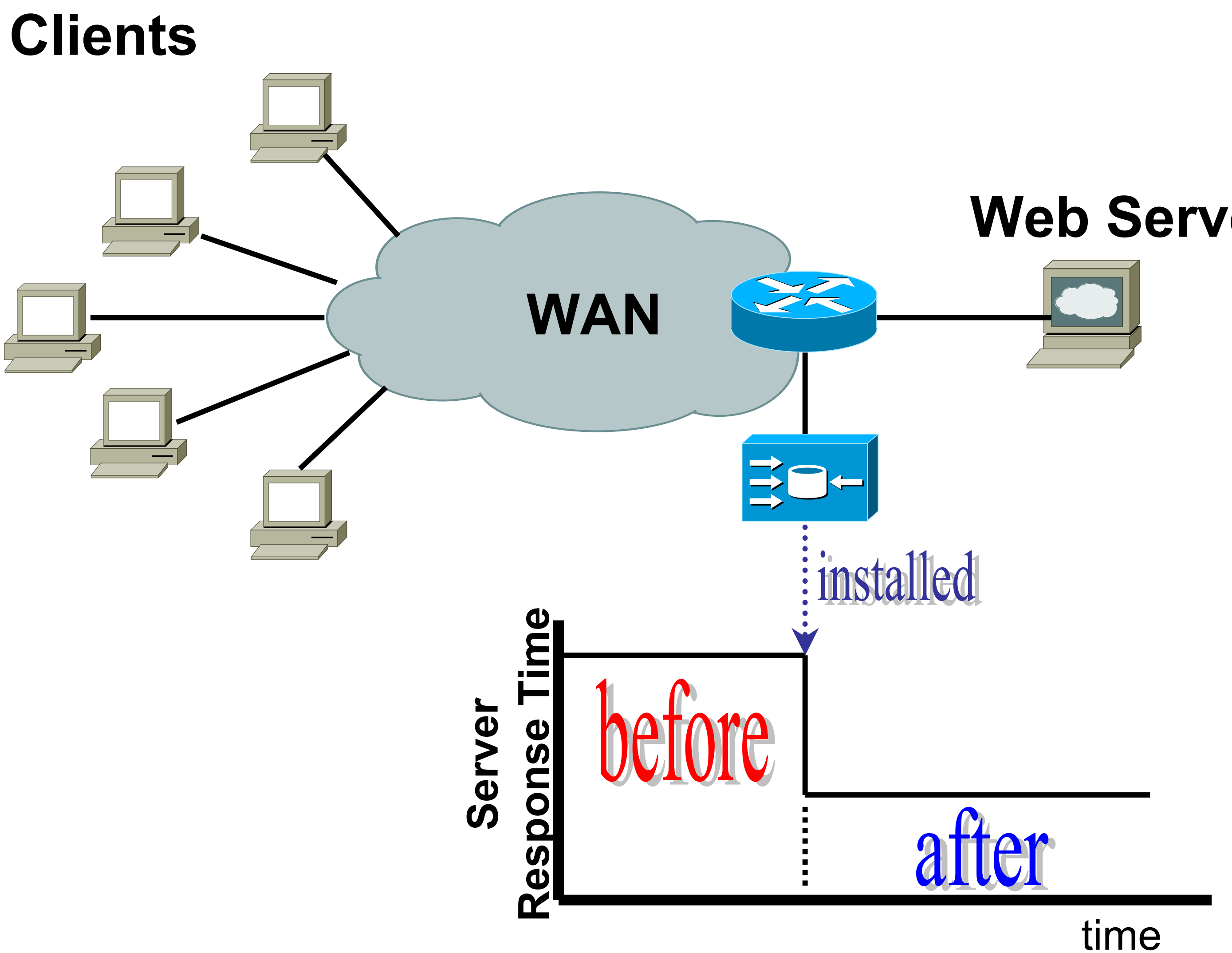
Cisco.com

Special Case: Web Server Protection

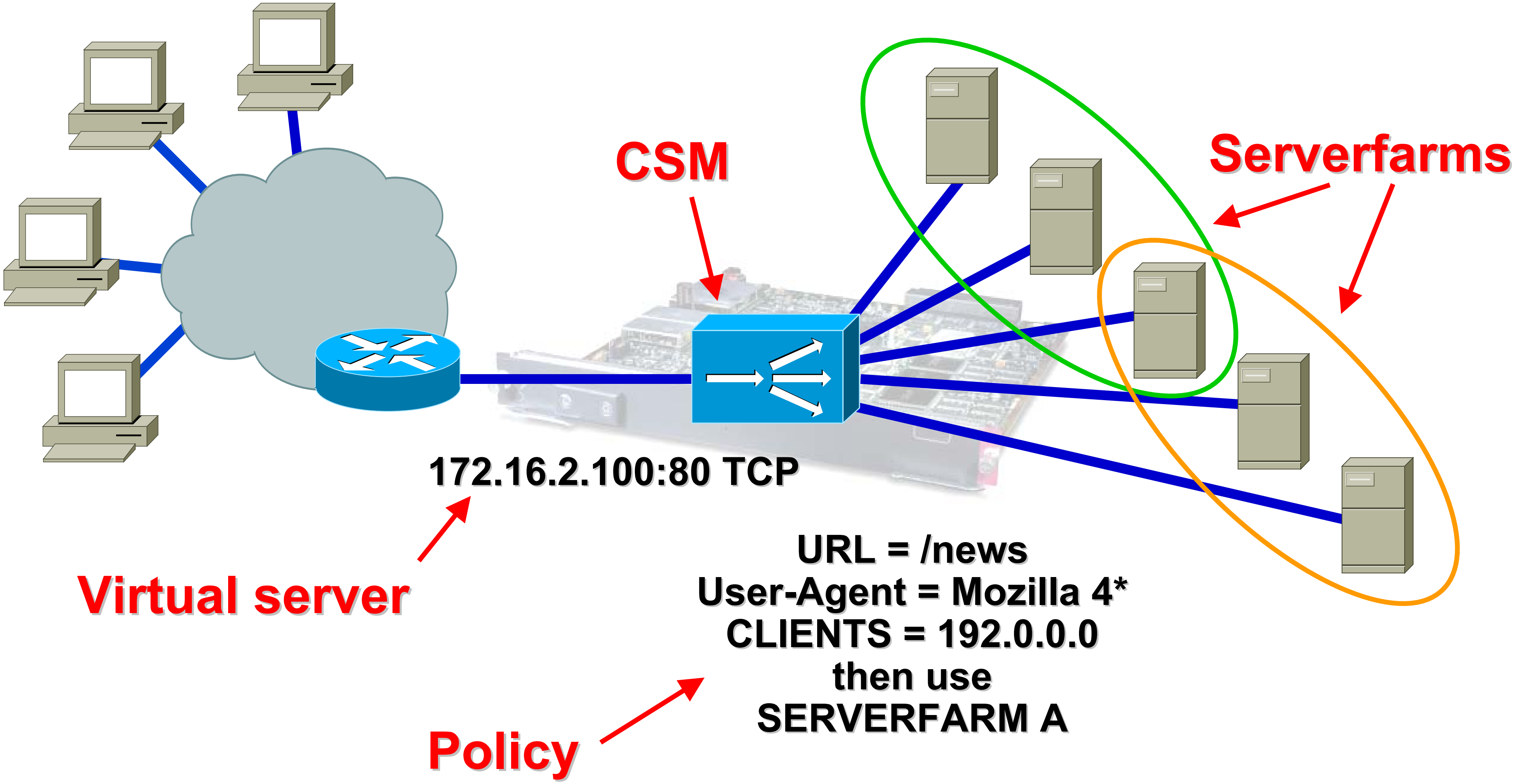
Web Server Protection

- **Heard about big web attacks last year?**
  No?
  Why not???
- **There is a solution: Content Networking**

Reverse Proxy Caching

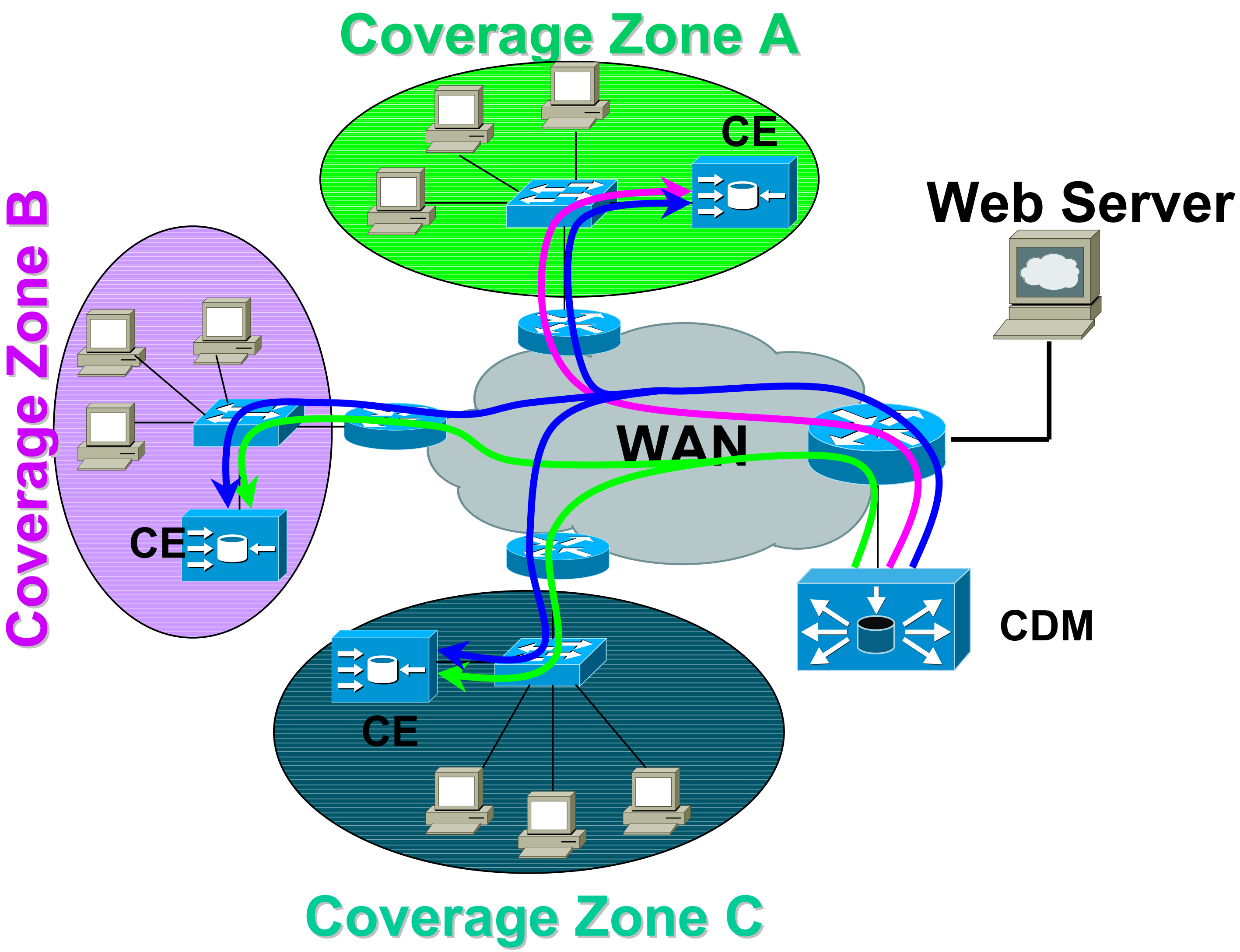


The Content Switching Module (CSM) for the Catalyst 6500



Built-in: TCP Intercept against SYN attacks!!

Content Distribution Networks



Cisco.com

Additional Solutions: Arbor and Riverhead

Cisco AVVID Partner Program

The Need for More…

Cisco.com

- On today's routers you have limited CPU
    - More "intelligence" needed
- Blocking, rate-limit mostly too course
    - Also affects "legal" traffic

➡ **Need a way to separate "good" from "bad" packets**

Arbor Networks Peakflow DoS: Hardware

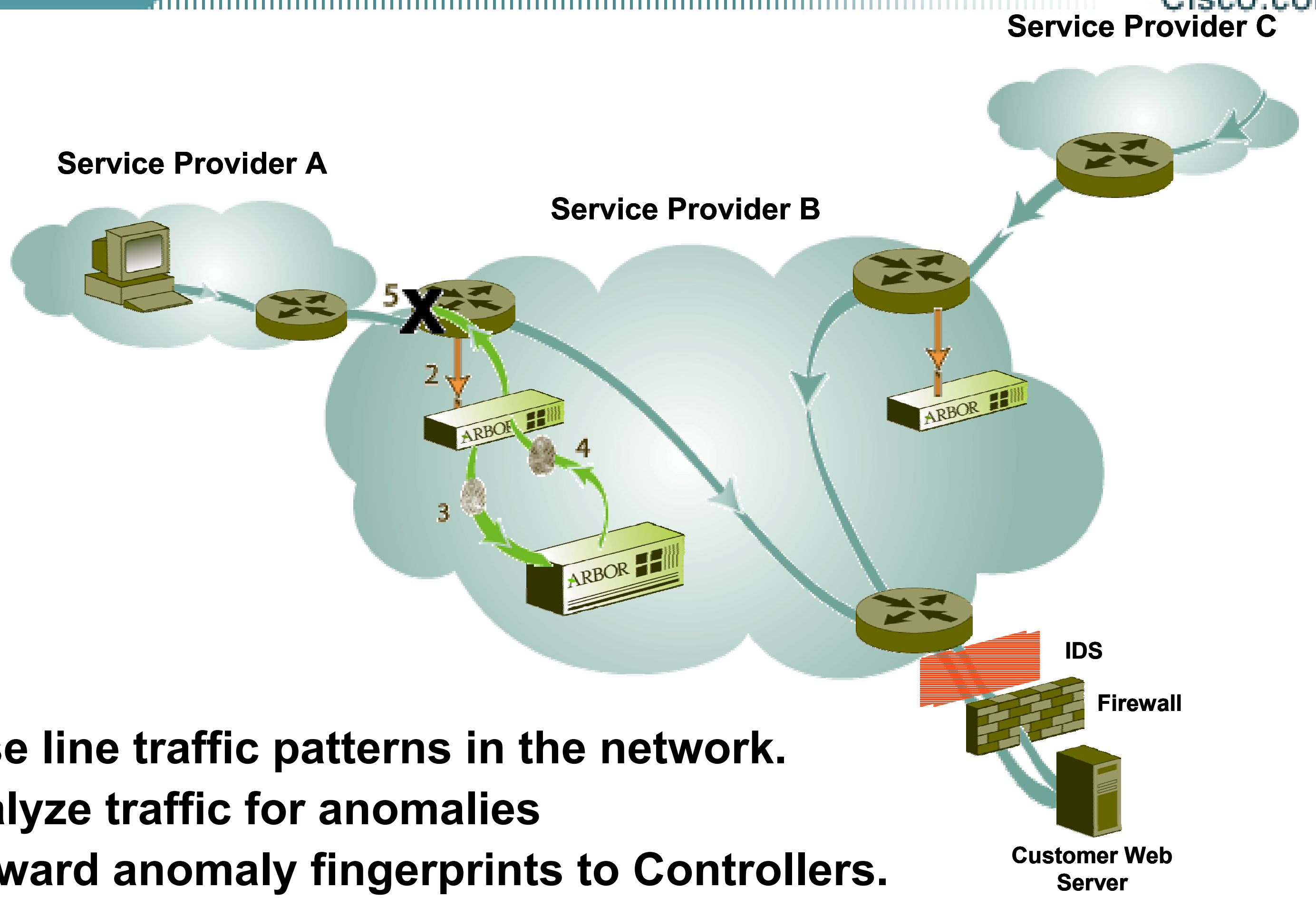
Collector

- 1RU rack height (1.7")
- Collects flow statistics from border and edge routers
- 2 types of collectors: NetFlow and Packet Capture
- Builds dynamic traffic baseline to detect bandwidth anomalies
- Works with Cisco routers and IDS/firewalls

Controller

- 2RU rack height (3.3"),
- Aggregates distilled anomaly data from Collectors
- Correlates distributed events to create network-wide view of all DoS attacks

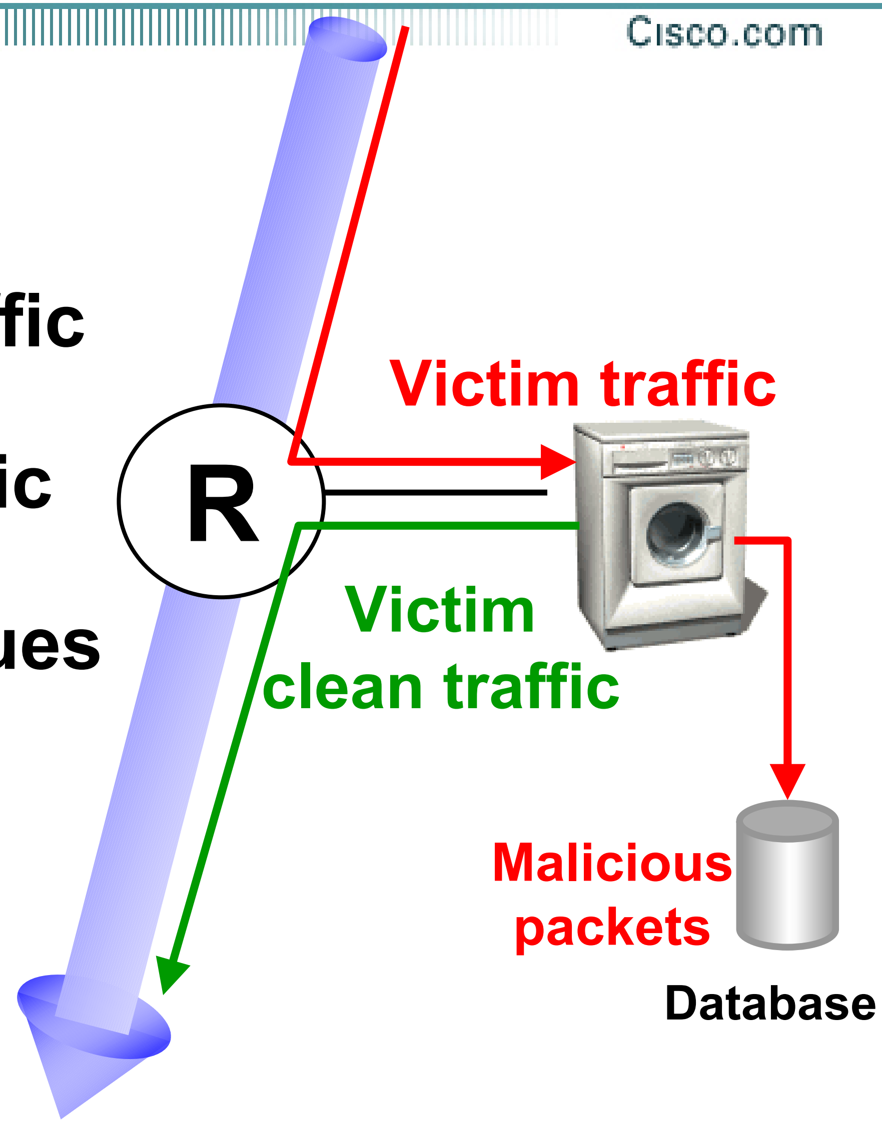
Arbor Networks: SP solution



1. **Profile** : Base line traffic patterns in the network.
2. **Monitor** : Analyze traffic for anomalies
3. **Detect** : Forward anomaly fingerprints to Controllers.
4. **Trace** : Trace the attack to its source.
5. **Filter** : Recommends filters (X)

Riverhead: Basic Concepts

1. Detection
2. Diversion of victims' traffic
3. Sieve out malicious traffic
4. Legitimate traffic continues on its route



Cisco.com

Wrap Up

What we can do now

Cisco.com

- Detect DoS Attacks (SNMP, NetFlow, ACL)
- Trace back random packet floods (NetFlow, ACLs, IP source tracker)
- Shun a source (uRFP, ACL)
- Shun a destination (routing, ACL)
- Limit attacking traffic (CAR, PIRC)
- Remote trigger via iBPG
- Protect Web servers (CSM / Content Networking)
- Understand partner solutions (Arbor, Riverhead)

Tip: scheduler allocate

Cisco.com

- Schedules CPU time spent on processes versus interrupts

Syntax:

```
scheduler allocate <interrupt> <processes>
```

<interrupt>: **3000-60000 Microseconds handling network interrupts**

<processes>: **1000-8000 Microseconds running processes**

Example:

```
router(config)#scheduler allocate 8000 8000
```

Very useful under heavy load!
Recommended Standard Config!

And Most Important:

Be Prepared!!!!!

- **Most tricks need pre-configs**
- **Install a black hole router!**
- **Learn what you can / cannot do (ACLs!)**
- **Practise, practise, practise, …**

Other Complementary Sessions

Cisco.com

SEC-211:	Security on Routers
SEC-307:	Security on Ethernet Switches
SEC-200:	Network Security: Risk and Threat model
SEC-201:	Network Security: Design and Attack Mitigation
SEC-204:	Understanding and Deploying Intrusion Detection Systems
SEC-214:	The security of MPLS VPN

References (non-Cisco)

Cisco.com

DoS Detection:

- “Tackling Network DoS on Transit Networks”: David Harmelin, DANTE, March 2001 (describes a detection method based on NetFlow) [http://www.dante.net/pubs/dip/42/42.html]
- “Inferring Internet Denial-of-Service Activity”: David Moore et al, May 2001; (described a new method to detect DoS attacks, based on the return traffic from the victims, analysed on a /8 network; very interesting reading) [http://www.caida.org/outreach/papers/backscatter/index.xml]
- “The spread of the code red worm”: David Moore, CAIDA, July 2001 (using the above to detect how this worm spread across the Internet) [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.enteract.com/~robt/Docs/Articles/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]

References (Cisco)

Cisco.com

Product Security:

- Cisco's Product Vulnerabilities; A page that every SE MUST know!!! [http://www.cisco.com/warp/public/707/advisory.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 [http://www.cisco.com/public/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]

Mailing lists:

- cust-security-announce: All customers should be on this list.
- cust-security-discuss: For informal discussions.

Surviving a DoS Attack

SEC-301

Please Complete Your Evaluation Form

SEC-301

CISCO SYSTEMS

EMPOWERING THE INTERNET GENERATION℠