Securing a Core Network

Manchester, 21 Sep 2004

Michael Behringer <mbehring@cisco.com>

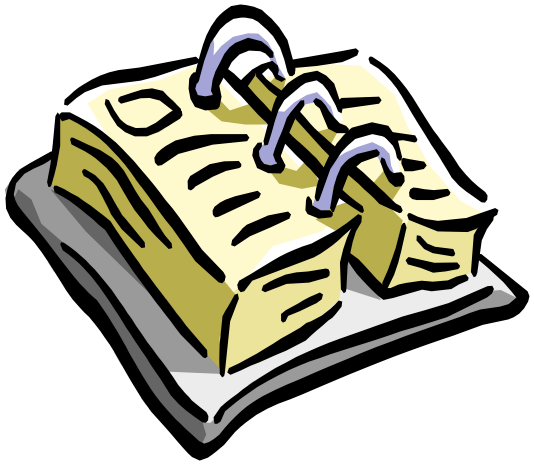
Christian Panigl <panigl@noc.ACO.net>

Attacks Against Routers

- **Everyday attacks**
  - **Router compromise (0wn3d) due to weak password**
  - **TCP amplification/reflection attacks**
  - **Packet floods**
  - **Receive path saturation**
- **New**
  - **IP options**
  - **IPv6 attacks**
  - **“Giga-Bots”: More than 1 Gbit/s attack traffic!**

Core Security More Important than Ever!

Securing a Core Network Agenda



0915 Overview on Securing a Core

- **securing a router, routing security, management security, ingress filtering (RFC2827)**

1030 break

1100 Core hiding techniques

- **ISIS, private address space, MPLS**

1130 Infrastructure ACLs

- **What it is, How to deploy, pros and cons**

1200 Discussion

1230 end

Overview on Securing a Core

Core Security Overview

1. **Basic Security**

   AAA, SSH, SNMPv3, rACL, CoPP, etc…

   → **Individual router security**

2. **Don't let packets into (!) the core**

   → No way to attack core, except through routing, thus:

   → **Still "open": routing protocol**

3. **Secure the routing protocol**

   Neighbor authentication, maximum routes, route filters, dampening, GTSM, …

   → **Only attack vector: Transit traffic**

4. **Design for transit traffic**

   Correct Core Design (Capacity / QoS)

   Bogon filters (RFCs 2827, 3330, 3704)

   Choose correct router for bandwidth

   → **Now only insider attacks possible**

5. **Operate Securely**

   → **Avoid insider attacks**

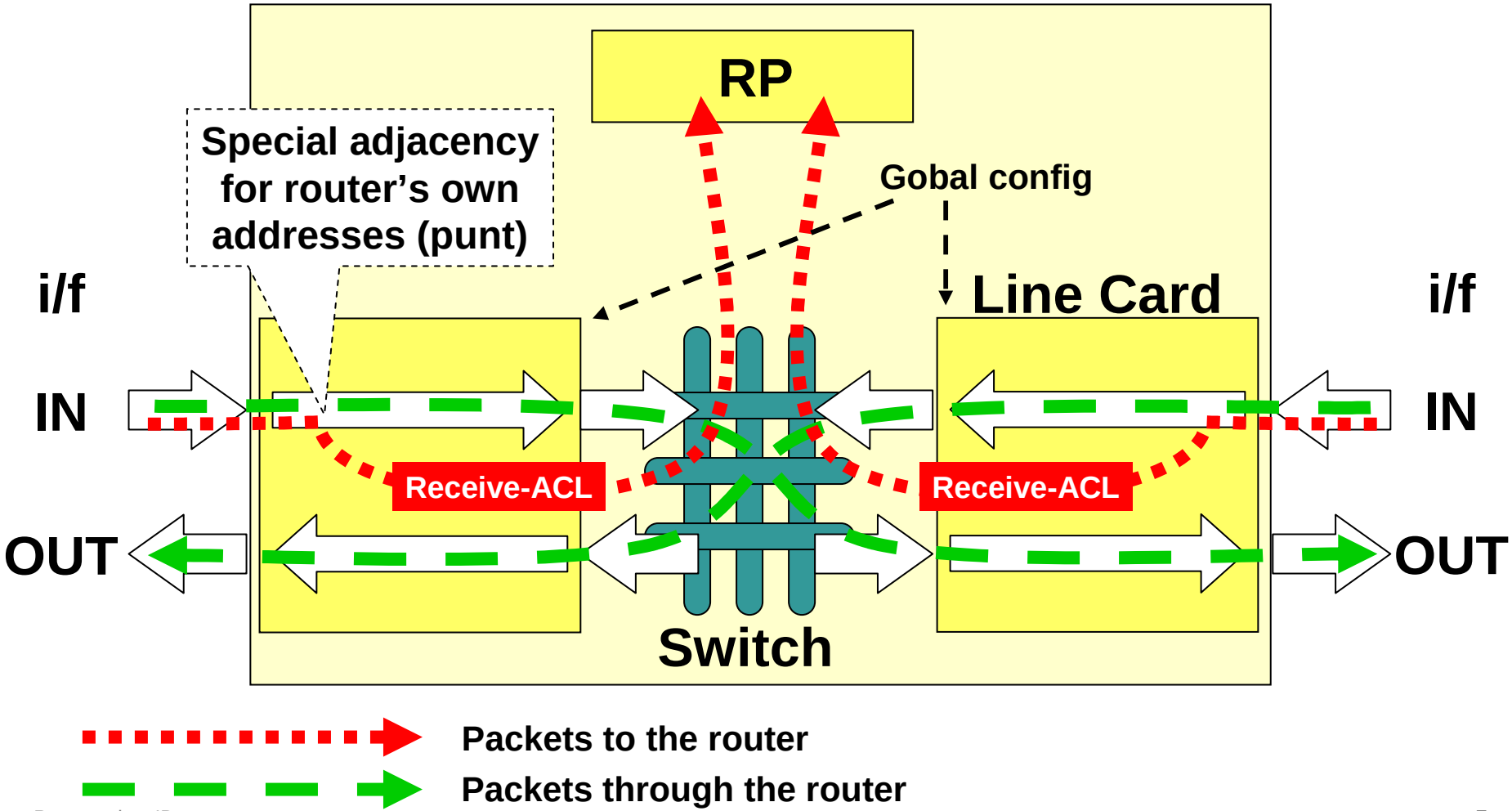
Router Security Basics

- AAA: Central admin control, logging, etc.
- SSH instead of Telnet
- SNMPv3 for managment

… plus many security best practices

Receive ACLs

[no] ip receive access-list <num>



Receive ACL: Features

- **Single ACL, for *all* interfaces**
- **Only checks traffic *to* the router (receive traffic)**
- **Does not affect transit traffic**
- **In short:**

  permit traffic that needs to get to the router (routing, NTP, SSH, …)

  deny the rest

Control Plane Protection (CoPP)

- More powerful than rACLs
- In addition to permit / deny allows rate limits to control plane
- Will be on all platforms
- Will eventually replace rACLs

Core Security Overview

1. **Basic Security**

   AAA, SSH, SNMPv3, rACL, CoPP, etc…

   → **Individual router security**

2. **Don't let packets into (!) the core**

   → No way to attack core, except through routing, thus:

   → **Still "open": routing protocol**

3. **Secure the routing protocol**

   Neighbor authentication, maximum routes, dampening, GTSM, …

   → **Only attack vector: Transit traffic**

4. **Design for transit traffic**

   Correct Core Design

   Capacity / QoS

   Choose correct router for bandwidth

   → **Now only insider attacks possible**

5. **Operate Securely**

   → **Avoid insider attacks**

Infrastructure ACLs

- See second part of this presentation
- (left separate for discussion)

Core Security Overview

1. **Basic Security**

   AAA, SSH, SNMPv3, rACL, CoPP, etc…

   → **Individual router security**

2. **Don't let packets into (!) the core**

   → No way to attack core, except through routing, thus:

   → **Still "open": routing protocol**

3. **Secure the routing protocol**

   Neighbor authentication, maximum routes, dampening, GTSM, …

   → **Only attack vector: Transit traffic**

4. **Design for transit traffic**

   Correct Core Design

   Capacity / QoS

   Choose correct router for bandwidth

   → **Now only insider attacks possible**

5. **Operate Securely**

   → **Avoid insider attacks**

Routing Security: Verify the Peer

- All routing protocols permit MD5 checks
- Some with key chains for smooth key roll-over
- Allows to identify the peer
- Does not protect against DoS!

Peer Authentication

- **RFC1321 describes MD5.**

  http://www.ietf.org/rfc/rfc1321.txt

- **draft-ietf-idr-md5-keys provides suggestions for building MD5 keys.**

  http://www.ietf.org/internet-drafts/draft-ietf-idr-md5-keys-00.txt

- **RFC2385 describes using MD5 with BGP.**

  http://www.ietf.org/rfc/rfc2385.txt

Routing Security: Verify the Prefixes

- **Prefix filters:**

  deny bogons, RFC1918, your own space.

  http://www.cymru.com/Bogons/

  Or: Permit only IRR "known" prefixes (more secure, but more work)

- **Filter on Prefix Length**

  Do not accept longer than /24 for example

Routing Security: Verify the AS

- **Filter on:**

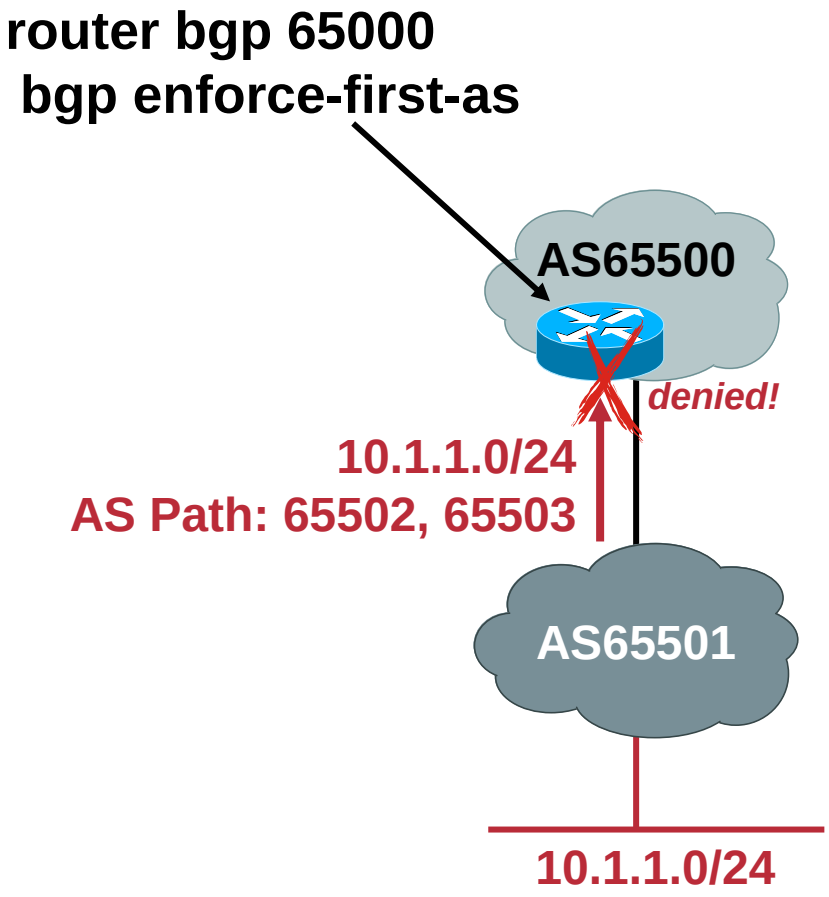
  AS of origin

  AS path (with regular expressions)

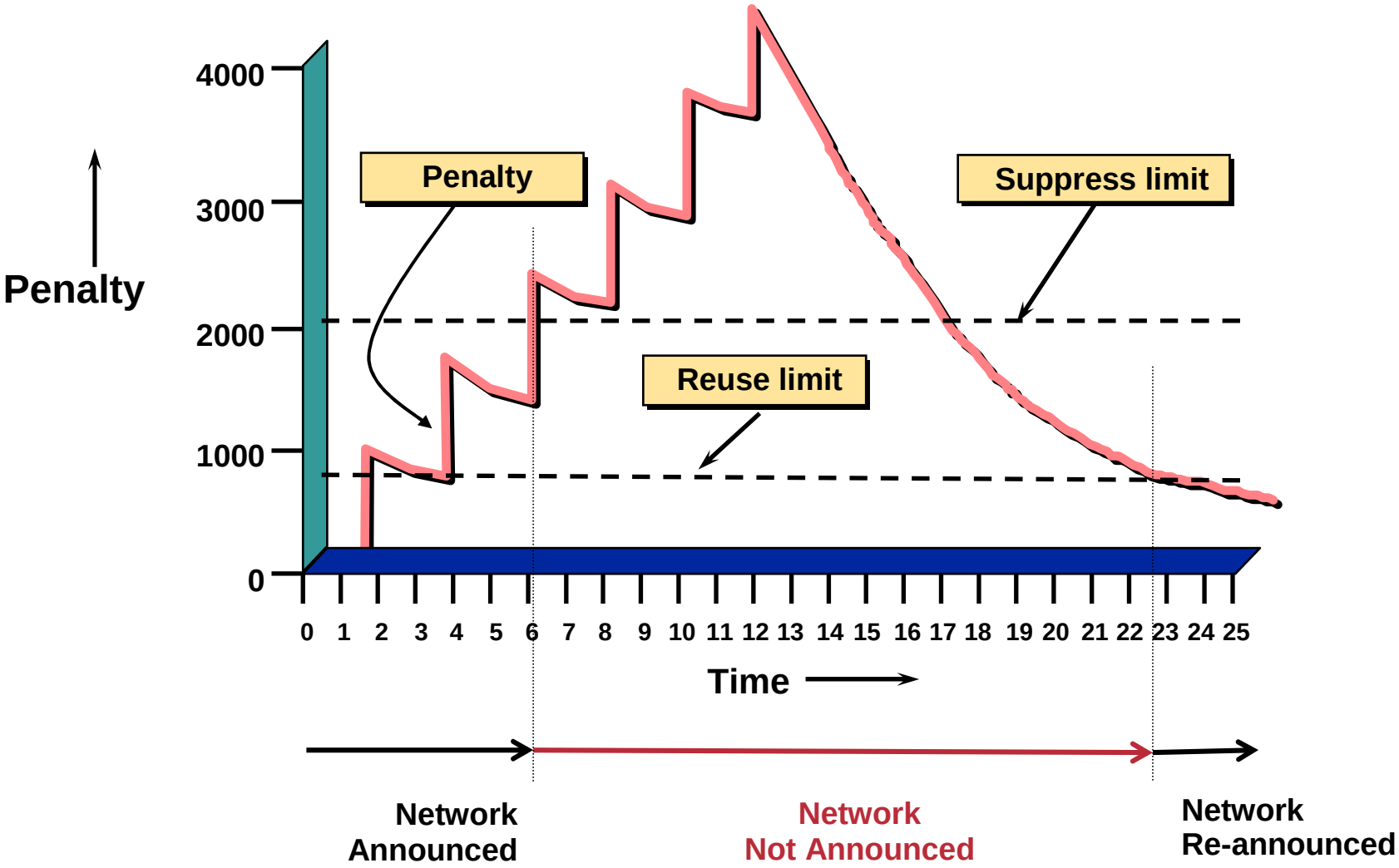
- **Filter as tight as possible**

Enforce First AS

- Prevents a BGP peer from advertising a route as if it is sourced from another autonomous system.
- Use this with all of your peers!
- CSCea00782 turns enforce-first-as on permanently.



BGP Damping



BGP Damping

- If prefix flaps, increase “penalty” counter
- If “penalty” > suppress_limit:
  Dampen prefix
- Continuously decrement “penalty”
- When “penalty” < re-use_limit:
  Re-use prefix

See RFC 2439

BGP Damping Configuration

Fixed damping

router bgp 100

bgp dampening [<half-life> <reuse-value> <suppress-penalty> <maximum suppress time>]

Selective and variable damping

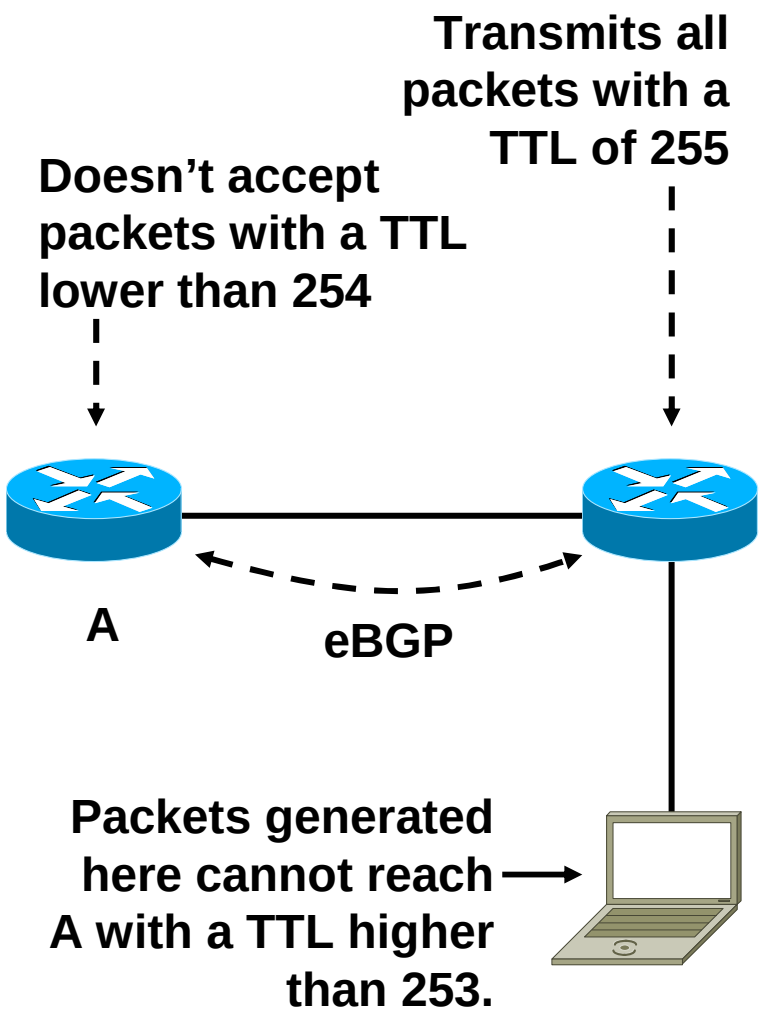
bgp dampening [route-map <name>]

Recommendations for ISPs

http://www.ripe.net/docs/ripe-229.html

Generalized TTL Security Mechanism (GTSM) from 12.0(27)S, 12.3(7)T

- Define IP hops to BGP peer (typical: 1)
- Check TTL on inbound BGP packets:
- If TTL < 255 - max_hops
- then drop BGP packet



See RFC 3682

Maximum Prefix Number

- **Define a maximum prefix number per peer:**

retry after 2 min

neighbor 192.0.2.2 maximum-prefix 100 80 restart 2

max prefixes

warning at ..%

- **To avoid being overrun with too many prefixes (memory issues)**
- **Keyword "warning-only" to *not* reset the session**

Routing Security: Overview

to verify the peer	MD5 authentication
to limit the number of prefixes	maximum-prefix
to define/filter prefixes	prefix filter lists
to filter out too long prefixes	prefix filter lists
to check first AS	enforce-first-AS
to check AS path	AS path filter
to prevent effect of flapping	BGP damping
to block remote exploits	Generalised TTL Security Mech.

Recommendation: Configure ALL checks!

Open Attack Vectors

- **Only one possible attack:**

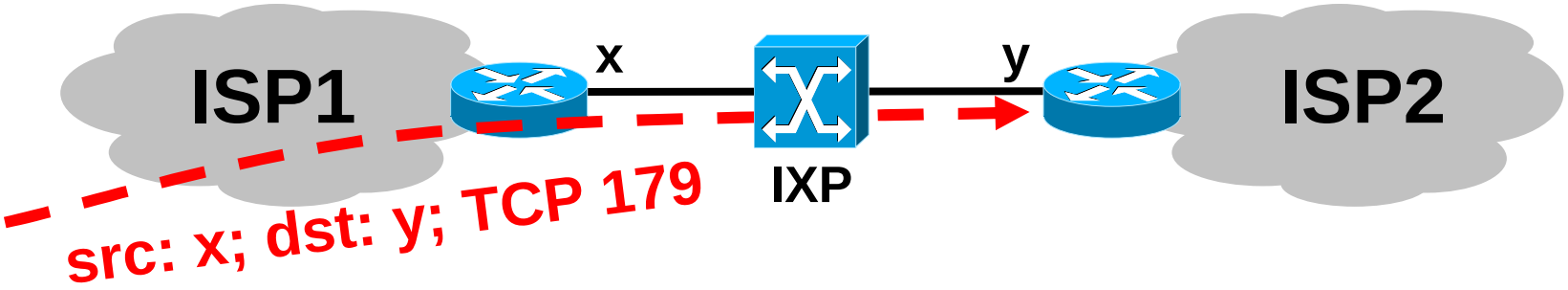
  DoS: spoofed source, to BGP peer

  Cannot fake/intrude, but might DoS BGP session

- **How easy is this?**

Attacking IXP Peerings

- IXP address spaces are known (IRR)
- → Easy to spoof BGP packets
- Can I get there?



- Not if: ISP 1 does anti-spoofing
- Not if: IXP address space not routed (and nobody defaults to either ISP, or ISPs don't default to IXP)

Core Security Overview

1. **Basic Security**

   AAA, SSH, SNMPv3, rACL, CoPP, etc…

   → **Individual router security**

2. **Don't let packets into (!) the core**

   → No way to attack core, except through routing, thus:

   → **Still "open": routing protocol**

3. **Secure the routing protocol**

   Neighbor authentication, maximum routes, dampening, GTSM, …

   → **Only attack vector: Transit traffic**

4. **Design for transit traffic**

   Correct Core Design

   Capacity / QoS

   Choose correct router for bandwidth

   → **Now only insider attacks possible**

5. **Operate Securely**

   → **Avoid insider attacks**

Latest Info on Address Ranges: Cymru

- www.cymru.com
- Lists of bogons, unused address space, etc.
- Use for ingress filters
- Regularly check for updates!!

Transit ACLs

- Normally: ISP Networks “permit ip any any” for transit
- “Transparency”
- Under extreme stress (worms, DoS):

  ISP apply temporary ACLs to filter attack/worm traffic
- Note: TEMPORARY
- Routers must support this

Discussion!

Re-Colouring at Edge

- Precedence 6&7: Reserved for routing
- No transit traffic should use prec 6 or 7
  - Problem with QoS on the core
  - Problem with routing protocols (same priority)
  - Routers look first at prec 6&7 traffic!!
  - → This can be a security risk
- Re-colour at edge!! (CAR)
- Depends on ingress line card / router

Discussion!

Three Golden Core Stability Rules

The Core must:

1. **be 100% stable**
2. **not be attackable**
3. **sustain DoS attacks across it**

Old network planning:

"normal" traffic + X

New network planning:

Ingress link full with 40 byte packets

Plan for DoS attacks to happen!!

Network Bandwidth Planning

- Assume: uplink completely loaded, 40 byte packets.
- Plan with this number: Can you handle

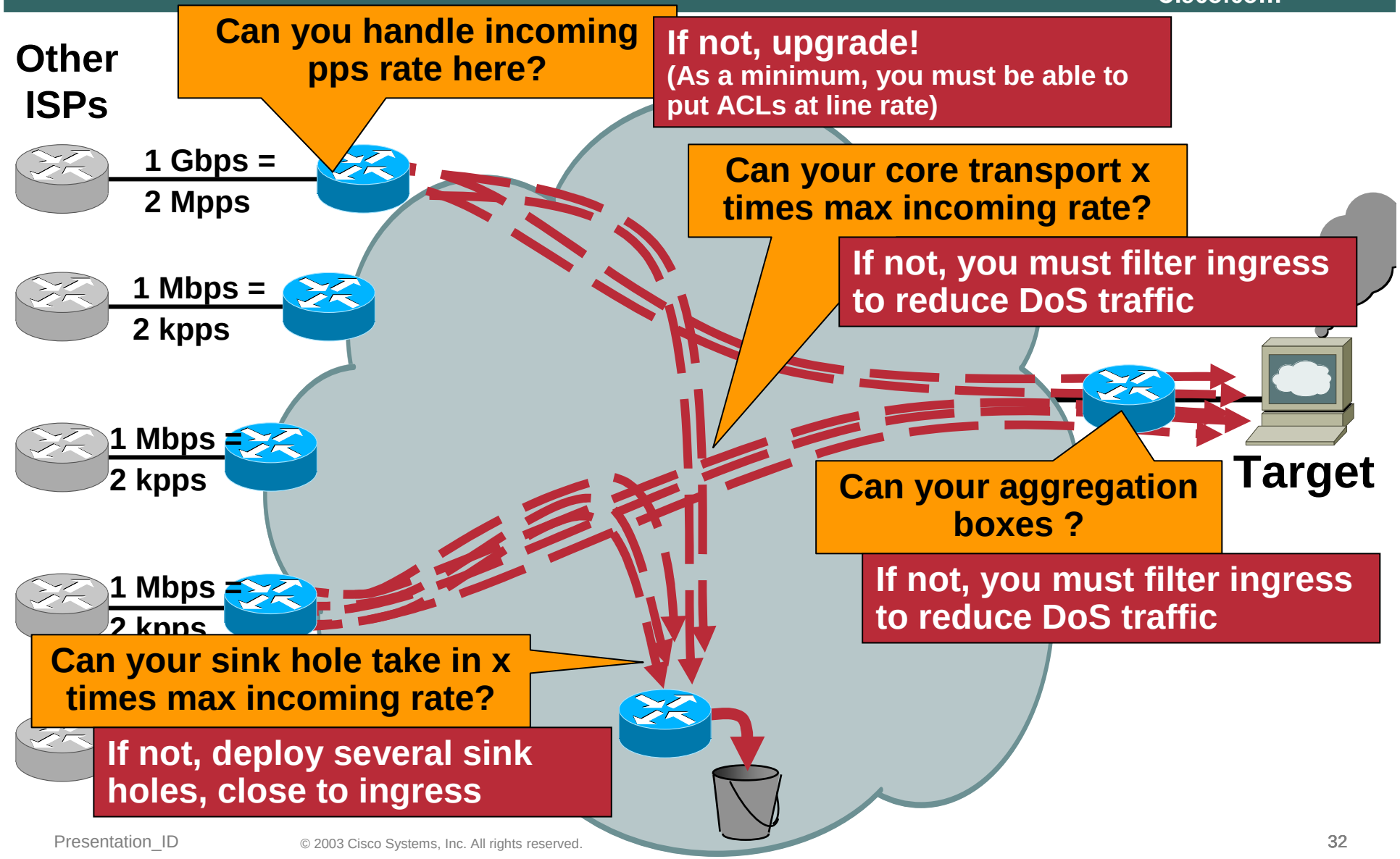
  ACLs, NetFlow, CAR, other features?

- Example: 1 GE uplink to an IXP

  1 Gbit/s / 8 bit/byte / 64 byte/packet ≈ 2 Mpps

- Can your ingress router handle that?
- Can your network?
- Can your sink-hole?
- What about floods on several ingress ports?

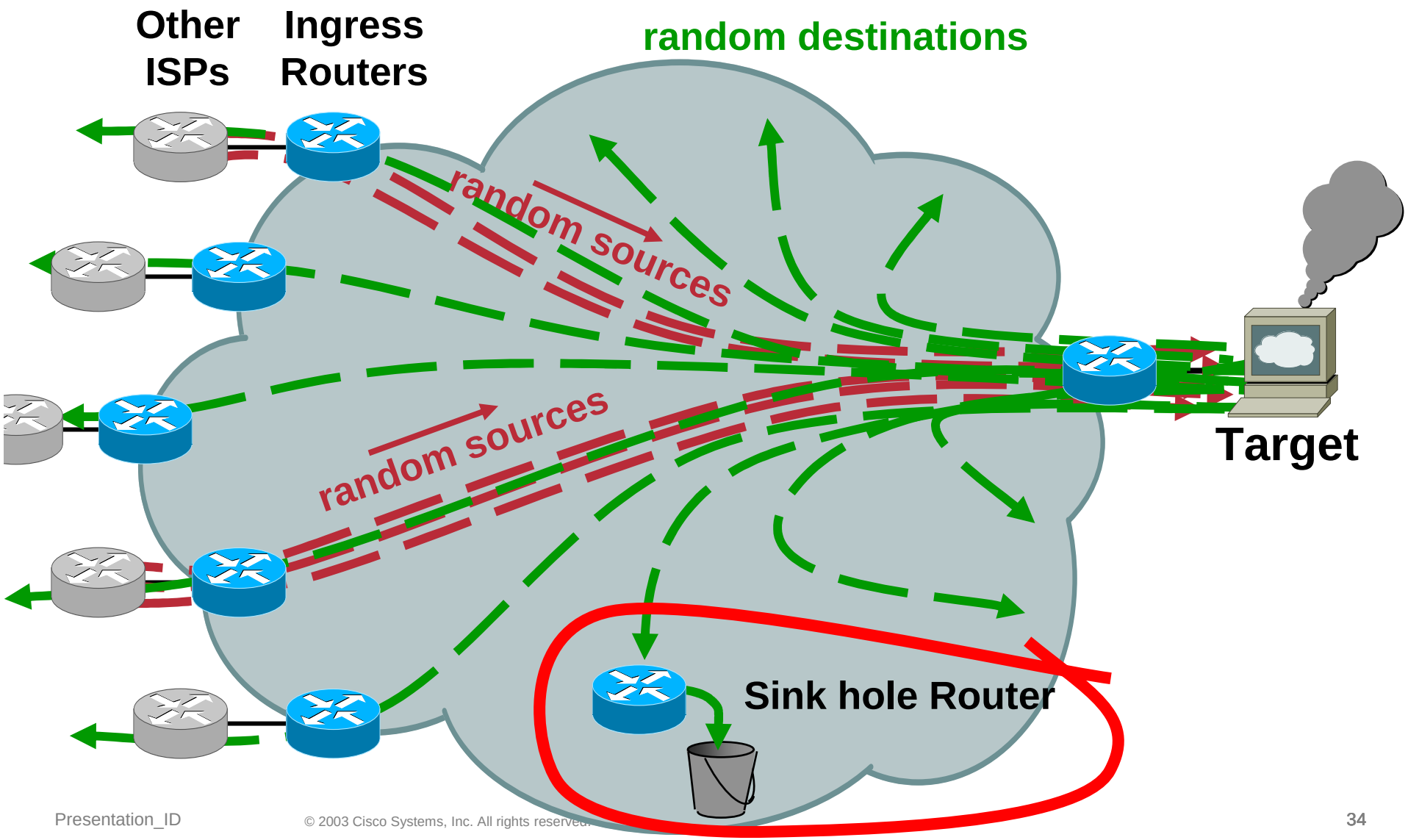
Bandwidth Planning: Assume DoS



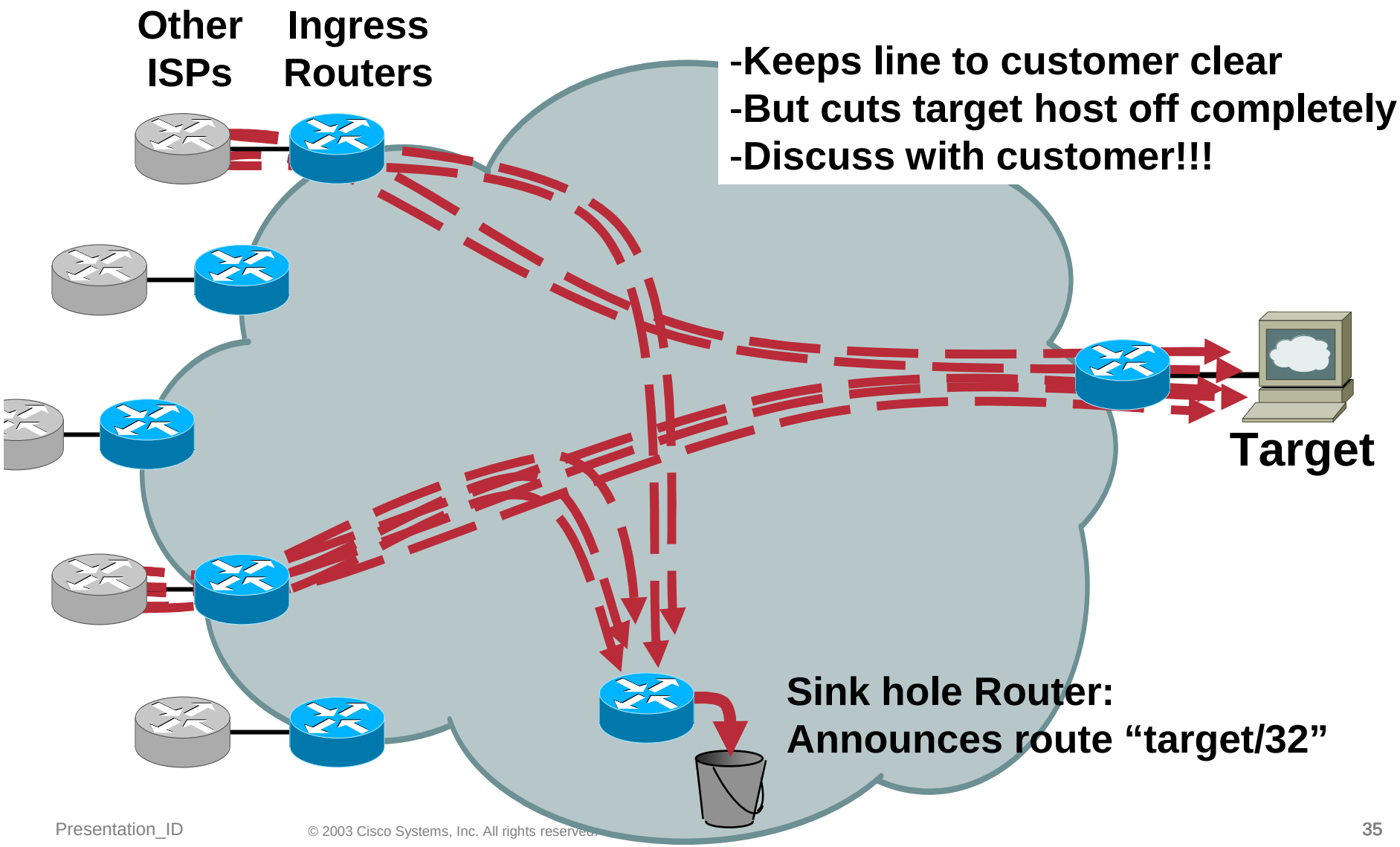
Sink Holes and Backscatter Analysis

- **Sink hole 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!!!**
- **Or, use default (if running full routing)**
- **Victim replies to random destinations**
- **-> Some backscatter goes to sink hole router, where it can be analysed**

Backscatter Analysis



Re-Directing Traffic from the Victim



Core Security Overview

1. **Basic Security**

   AAA, SSH, SNMPv3, rACL, CoPP, etc…

   → **Individual router security**

2. **Don't let packets into (!) the core**

   → No way to attack core, except through routing, thus:

   → **Still "open": routing protocol**

3. **Secure the routing protocol**

   Neighbor authentication, maximum routes, dampening, GTSM, …

   → **Only attack vector: Transit traffic**

4. **Design for transit traffic**

   Correct Core Design

   Capacity / QoS

   Choose correct router for bandwidth

   → **Now only insider attacks possible**

5. **Operate Securely**

   → **Avoid insider attacks**

Operational Security

- **Strict Permission Control**
  - **Who is allowed what**
- **Log every command, Sanity checks**
  - **Dual control: Network engineers have no access to log files**
- **Regular config checks**
  - **Specifically after re-boots**

Links

Lots of useful information at http://www.cymru.com/, specifically:

- **http://www.cymru.com/Documents/secure-ios-template.html**
- **http://www.cymru.com/gillsr/documents/junos-template.pdf**
- **http://www.cymru.com/Documents/secure-bgp-template.html**
- **http://www.cymru.com/gillsr/documents/junos-bgp-template.pdf**

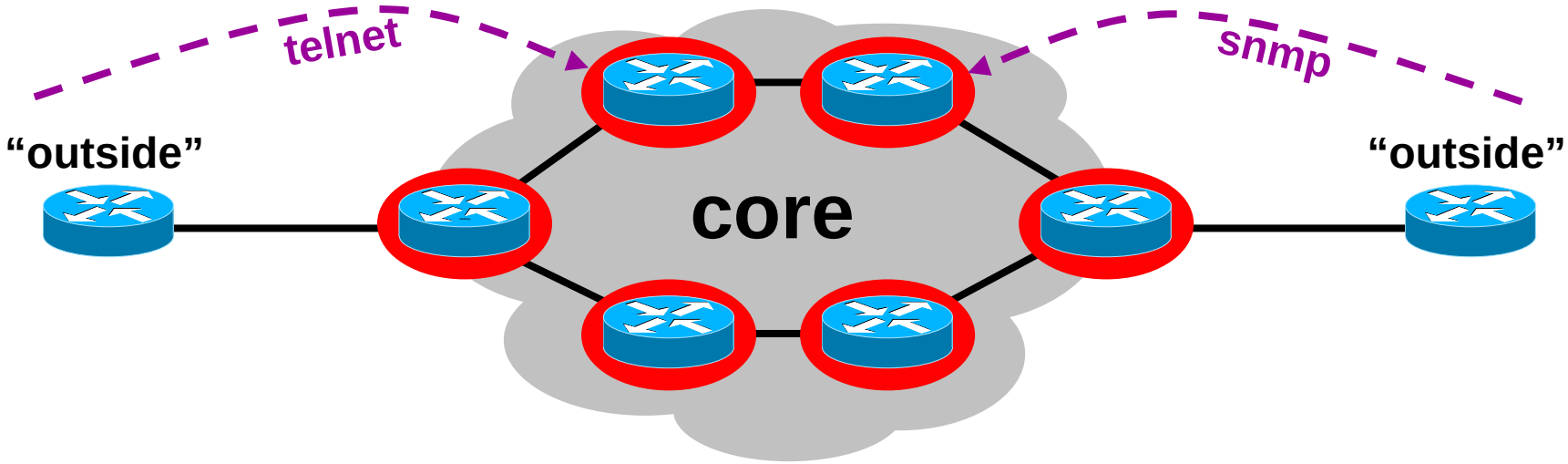
ISP Essentials:

Cisco Press book, by Barry Green and Philip Smith.

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

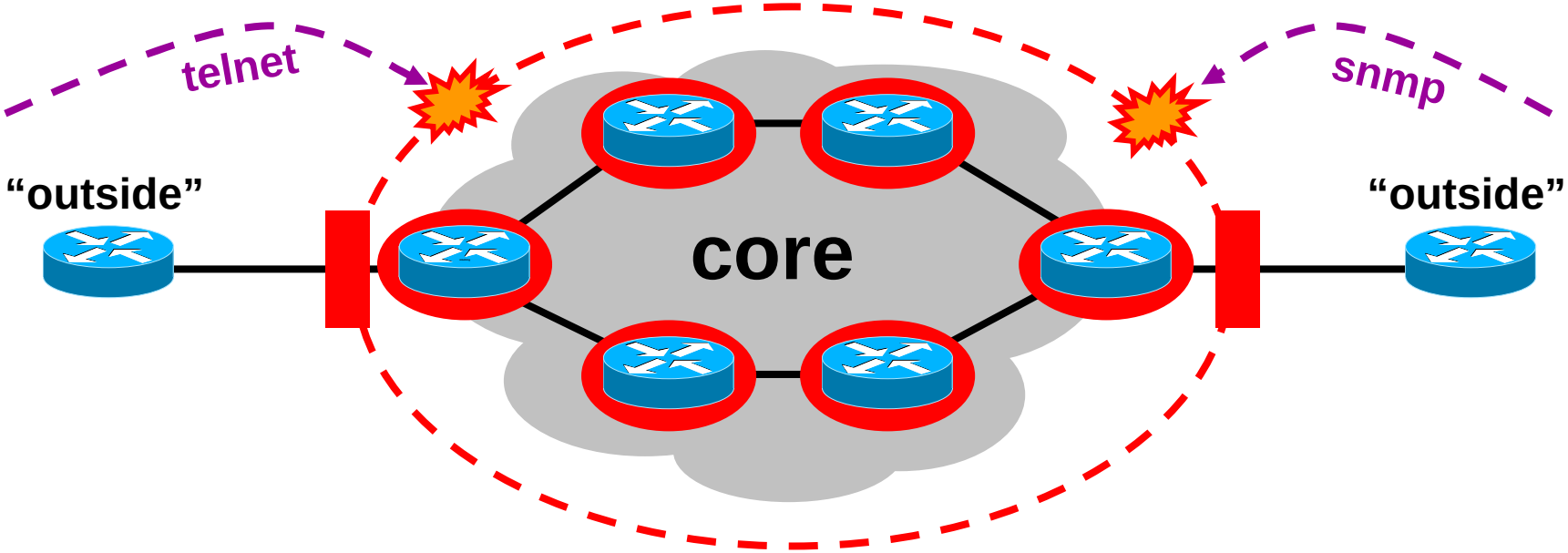
Core Hiding Techniques

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

Goal of this Presentation

- Some core security techniques have an impact on the global Internet
- Currently there is no commonly agreed “best current practice”
- Open discussion of pros and cons

Core Hiding Techniques

- Private Address Space
- Non-IP Control Plane
  - ISIS
- MPLS

Private Address Space (RFC1918)

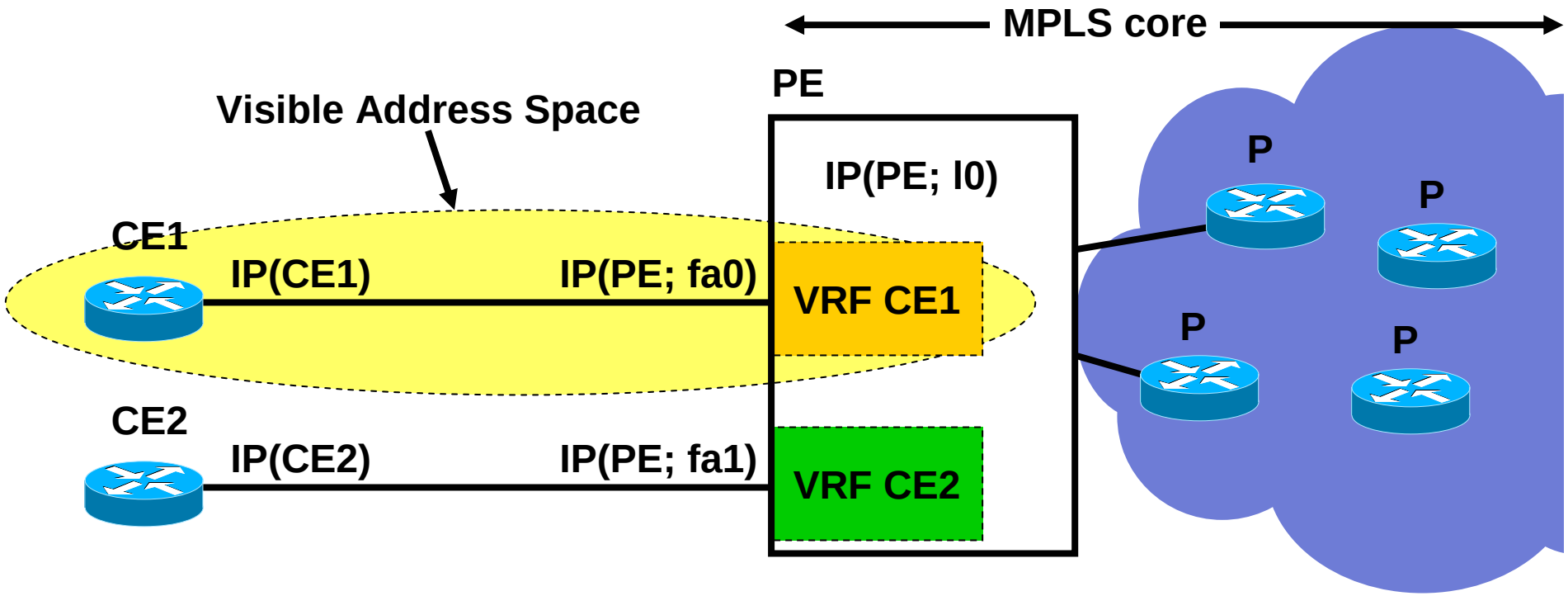
- All core interfaces get RFC1918 addresses
- All traffic from/to RFC1918 addresses blocked at ingress (implicit protection of core) => core interface addresses unreachable from outside core
- Blocking of traffic to edge interfaces (peering/upstream/customers) with non-private IP addresses still needs explicit ACL
- Troubleshooting (ping/traceroute) harder or even impossible from/to core devices
- Traceroute through core work but doesn’t resolve IP addresses externally

Non-IP Control Plane (CLNS/ISIS)

- Use of nonIP addresses & routing protocol for whole core
- Only loopback interface gets (possibly private) IP address
- Doesn’t even need any filtering to block traffic to core interfaces
- Blocking of traffic to edge interfaces (peering/upstream/customers) with IP addresses still needs explicit ACLs
- Troubleshooting (ping/traceroute) harder or even impossible directly from/to core devices

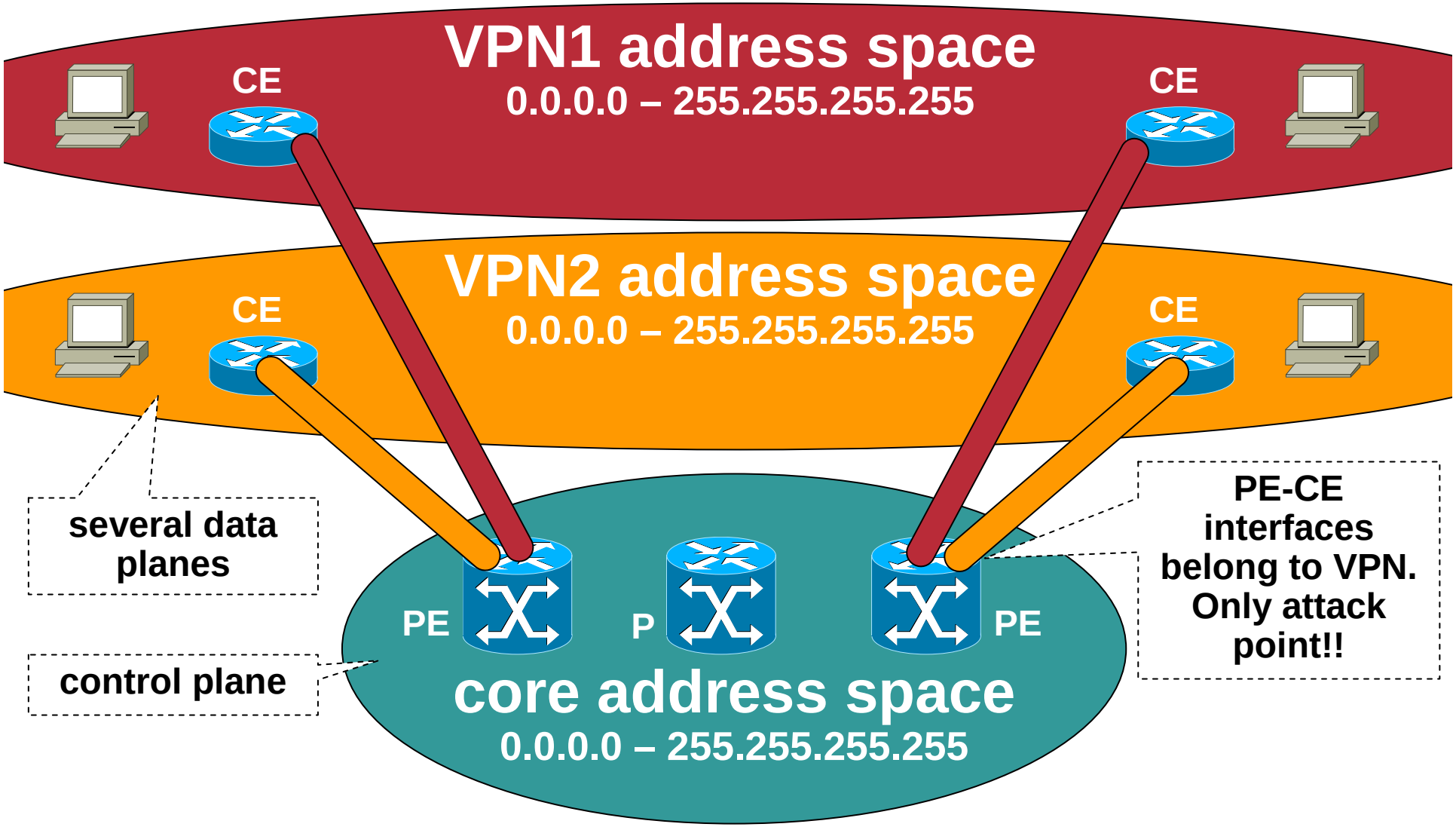
More Work Needed

Hiding of the MPLS Core Structure



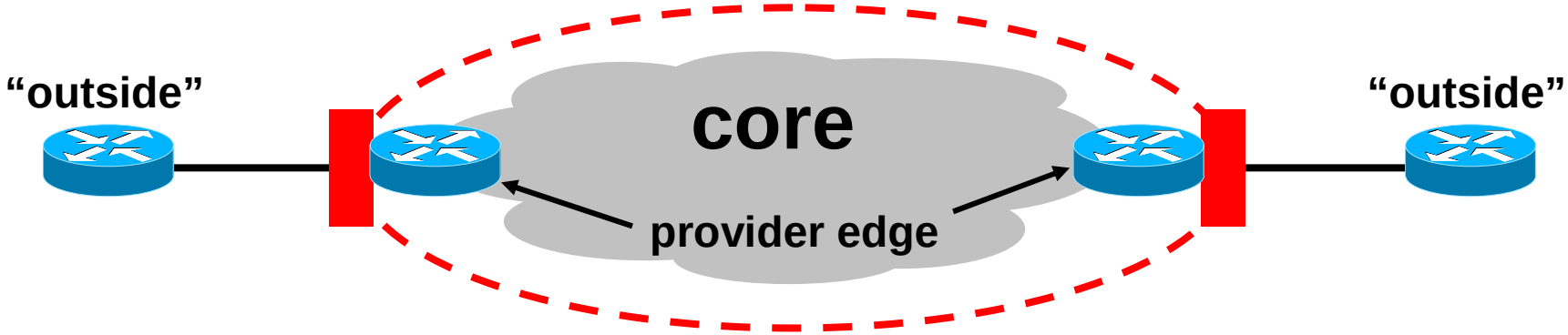
- VRF contains MPLS IPv4 addresses
- Only peering Interface (on PE) exposed (-> CE)!
  -> ACL or unnumbered

MPLS Core Hiding
Address Planes: True Separation!



Infrastructure ACLs

Securing the Core: Infrastructure ACLs



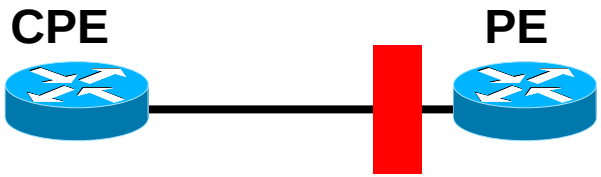
- On "PE": "deny ip any <core address space>"

  some exceptions, e.g. routing protocol from host to host
- Idea: No traffic to core → you can't attack
- Prevents intrusions 100%
- DoS: Very hard, only with transit traffic

Note: "PE" and "CE" are meant here as generic terms, not necessarily in the context of MPLS.

“Any packet directed towards your core network should be discarded – unless you are sure it is safe.”

Securing the Core: Infrastructure ACLs

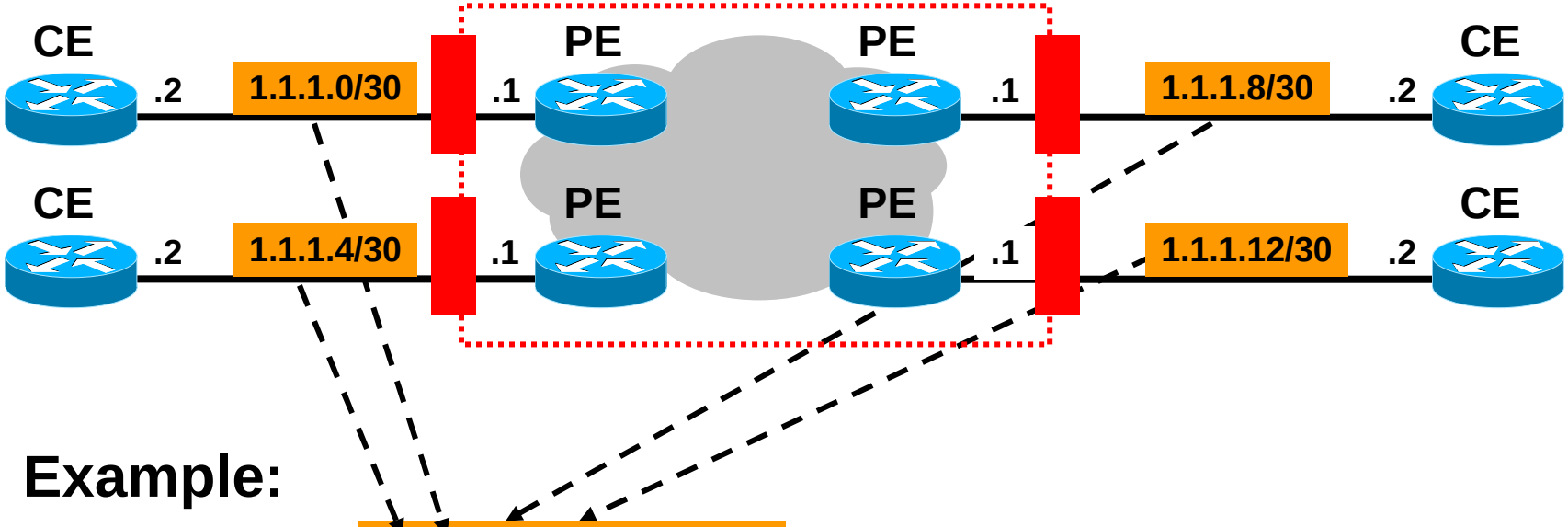


- On PE: “deny ip any <core address space>”

  Exception: Routing protocol from host to host

- Idea: No traffic to core → you can’t attack
- Prevents intrusions 100%
- DoS: Very hard, but traffic over router theoretically enables DoS.

Securing the Core: Infrastructure ACLs



- **Example:**

  deny ip any 1.1.1.0 0.0.0.255

  permit ip any any

- **Caution: This also blocks packets to the CE's!**

  Alternatives: List all PE i/f in ACL, or use secondary i/f on CE

Where to Apply Infrastructure ACLs?

- **To all ISP peers, downstream, upstream, peers:**
  - **On peering points on IXPs**
  - **On private peering points**
- **Towards customers**

To all "untrusted" networks, where attacks might enter the ISP core

Deployment

- **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**

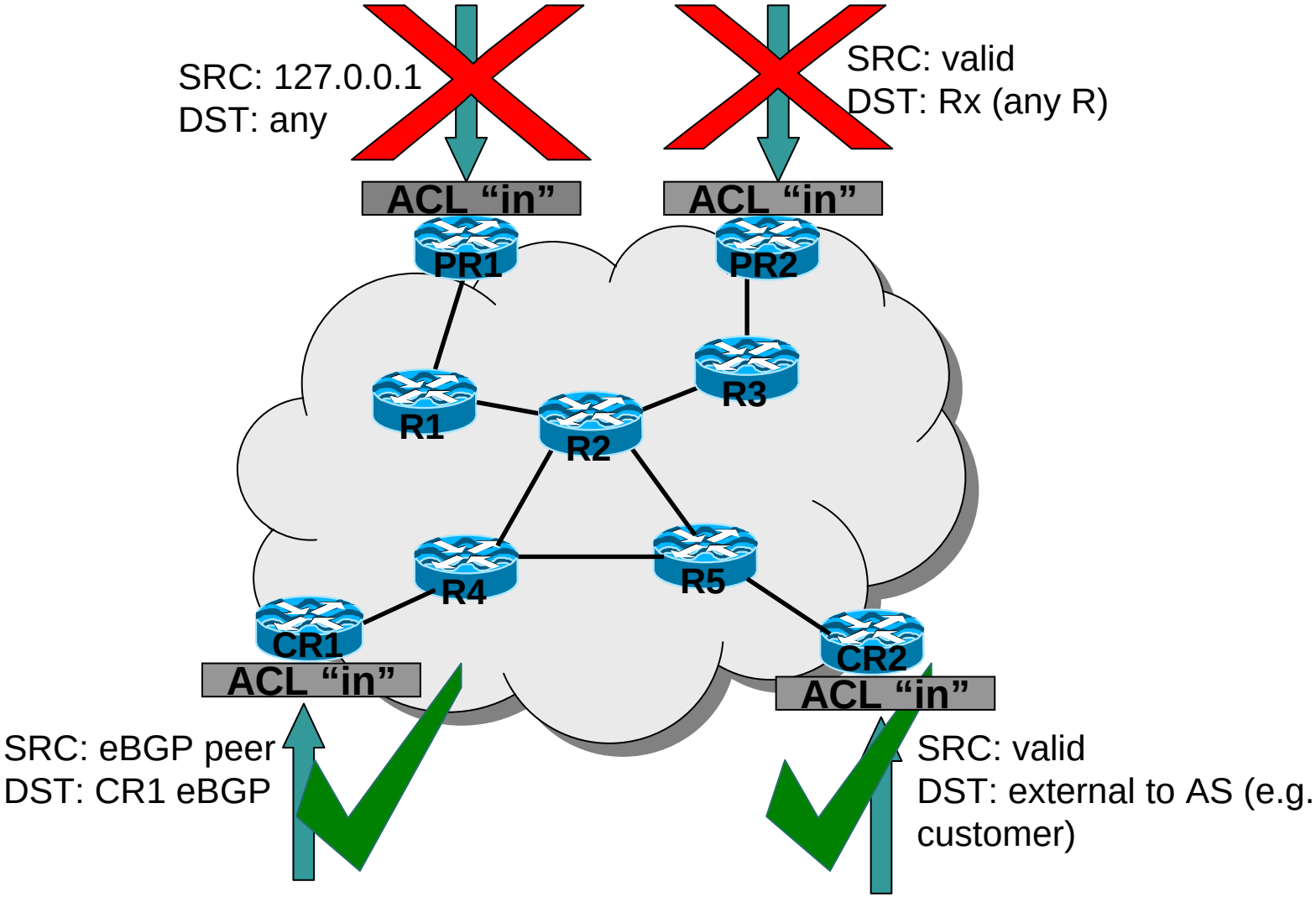
Deployment

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

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**
  - **Fragments pose a security risk: by default they are not filtered by ACLs**
  - **Fragments are likely not needed**
  - **access-list 101 deny/permit … *fragments***

Infrastructure ACL in Action



Step 1 – Which IP Protocols are Required?

- TCP – BGP, SSH, Telnet
- UDP – SNMP, NTP
- OSPF, EIGRP
- GRE
- ICMP to/from core routers
  - ICMP unreachables / TTL expired for traceroute
  - Do you require other ICMP? (e.g. ping)
  - Caution: ICMP can be used for DoS
- Caution: Only flows from “outside” your core

Step 1 – Classification ACL

- **Example:**

  permit tcp 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 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

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

! Deny src addresses of 0.0.0.0 and 127/8

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

! Deny RFC1918 space from entering AS

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

Example: Infrastructure ACL

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

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

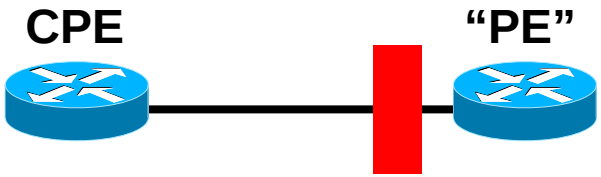
! Deny all other access to infrastructure

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

! Permit all data plane traffic

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

Reachability of CPE



- **Not always possible to block the subnet between PE and CPE:**

  If CPE is a PAT device

  If customer wants to ping CPE-CPE (although, could ping/trace from/to loopbacks, but requires enable mode for extended trace/ping)

Warning

- **Infrastructure ACLs are NOT a replacement for general network security**

  →Mistakes can happen

  →"Defence in Depth": Several layers of security

Secure routers, routing, management traffic as before!

rACL versus iACL

- iACL: Protection of the whole core
- rACL: Protection of a single router
- Edge routers should have BOTH rACL and iACL configured

Infrastructure ACLs: Pros

Security against:

1. **Operational mistakes (mis-configuration)**
2. **Bugs on the router (vulnerabilities)**

- **generally speaking, another layer of security around the core**

Infrastructure ACLs: Cons

1. Breaks transparency: Access from the outside through pings, traceroute *into* the core does not work. (Note: traceroute across n/w works!)
2. As a consequence, makes troubleshooting harder: from the outside, and from the core (traceroute from core routers to outside)
3. hard to deploy if core address space is not contiguous, or not easily expressed in an ACL
4. hardware does not support line speed ACLs on all platforms
5. hard to maintain (when core address space changes)

Discussion

Infrastructure ACLs: Bug or Feature?

Core Hiding: The right way forward?

ISP Essentials

- The “bible” for Core Security
- Available as book, and on FTP:

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

- How to secure the core

  Security for devices, routing, traffic, management, …