ISP and NSP Security Workshop

APRICOT 2011

Alcatel·Lucent

Free Use

- This slide deck can be used by any operator to help empower their teams, teach their staff, or work with their customers.
- It is part of the next generation of **APRICOT Security Curriculum ….** providing tools that can improve the quality of the Internet.

Goal

- Provide security core techniques/task that any SP can do to improve their resistance to security issues.
- These core techniques can be done on any core routing vendor's equipment.
- Each of these techniques have proven to make a difference.

Agenda

- Securing Infrastructure
- Gain Visibility
- MPLS / L3 VPN Security
- Understanding IPSec and SSL VPN
- Managed Security Services
- L2 Security and Attacks
- Conclusions

Time Table

- Day 1 / 2
  - Securing the Infrastructure
- Day 3
  - Gain Visibility
- Day 4 /5
  - MPLS / L3 VPN Security
- Day 5
  - Managed Security Services
  - L2 Security
- Conclusions

What Is Core Security?

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

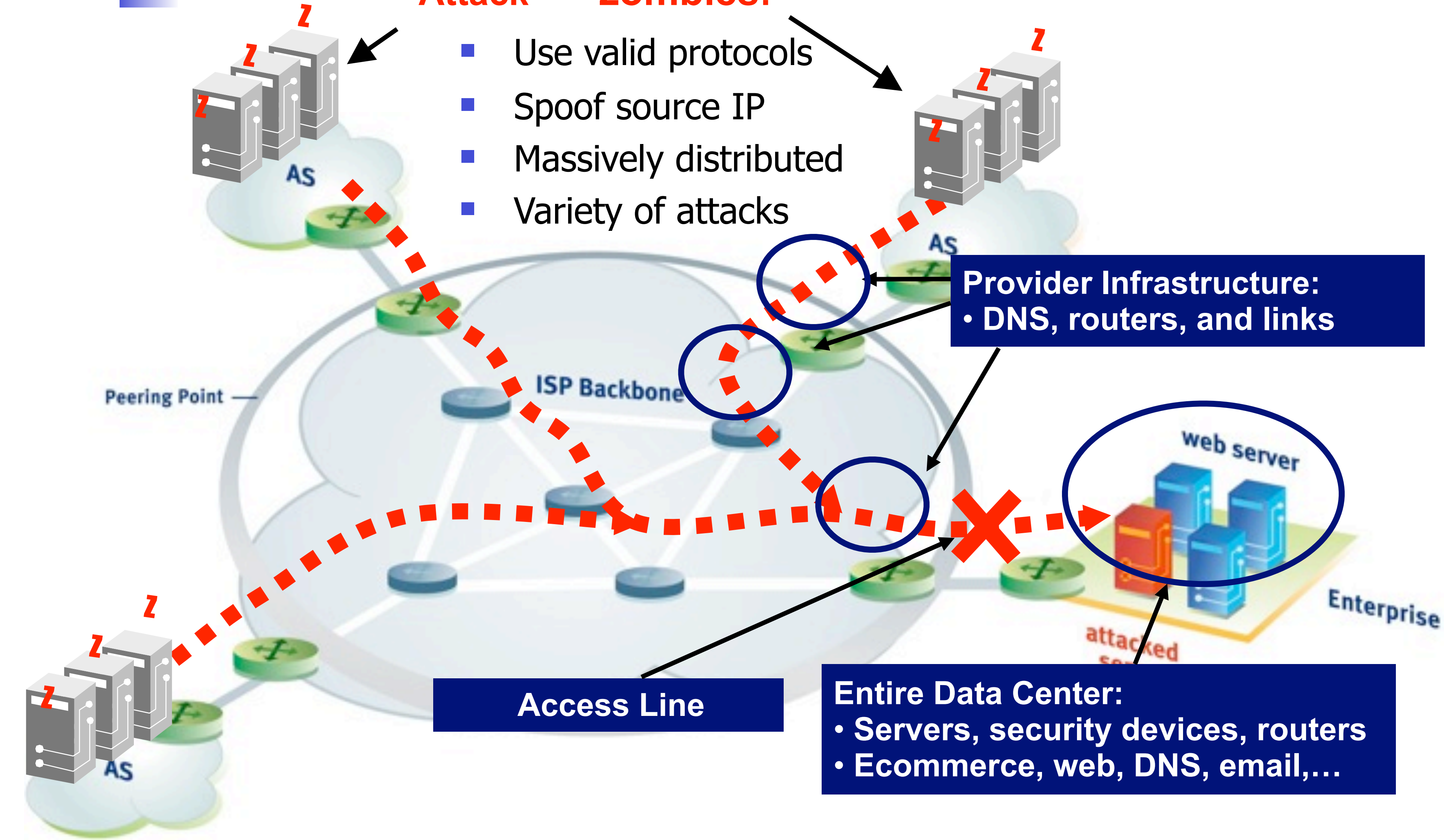
Denial of Service Attacks

- We understand intrusions (patch, patch, patch ;-))
- What about DoS? Do "the right things" and still suffer
- The vast majority of modern DoS attacks are distributed
  - DDos IS DoS
- DoS is often driven by financial motivation
  - DoS for hire :-(
  - Economically-driven miscreant community
- DoS cannot be ignored; your business depends on effective handling of attacks

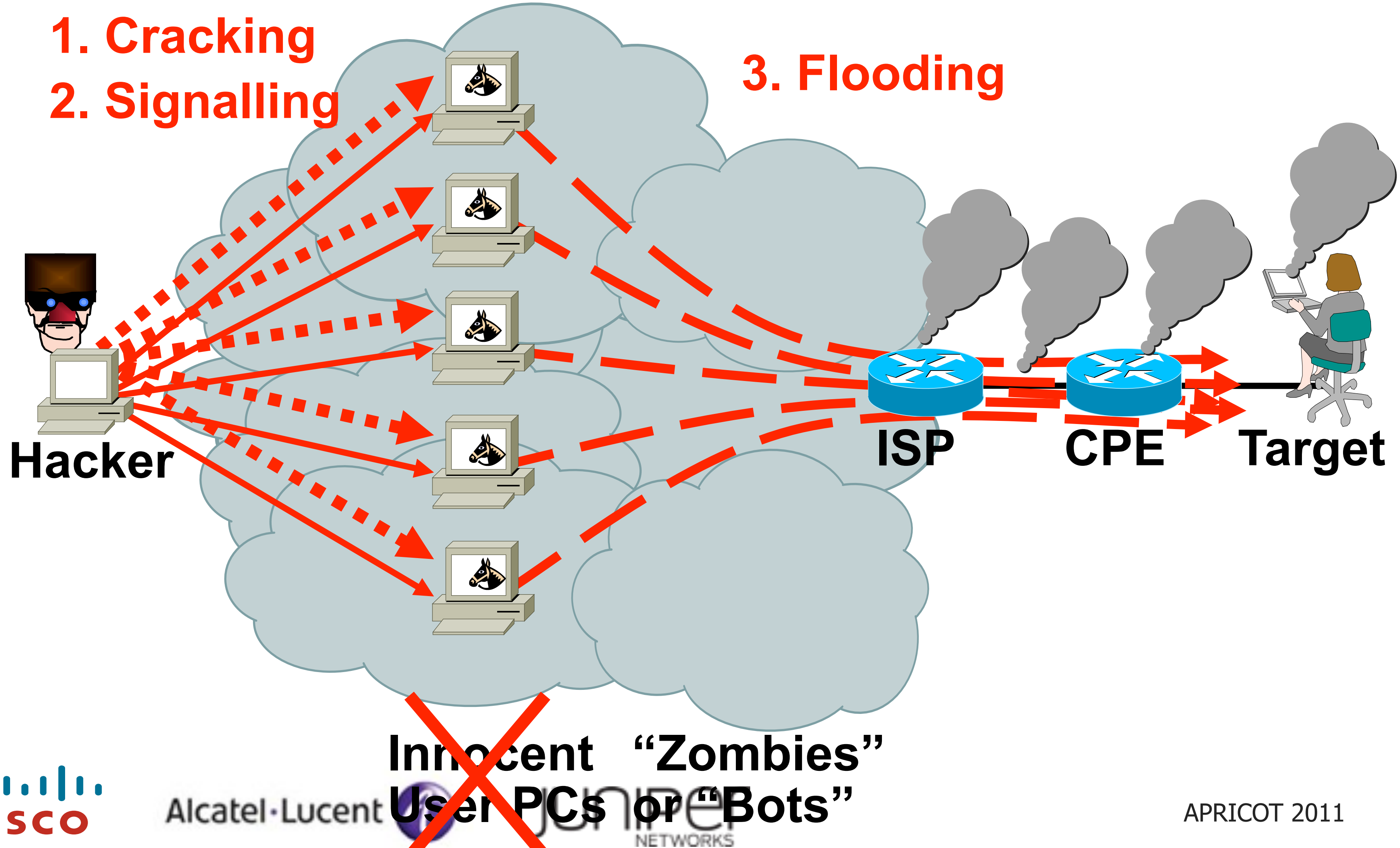
DDoS Vulnerabilities, Threats and Targets

Attack zombies:

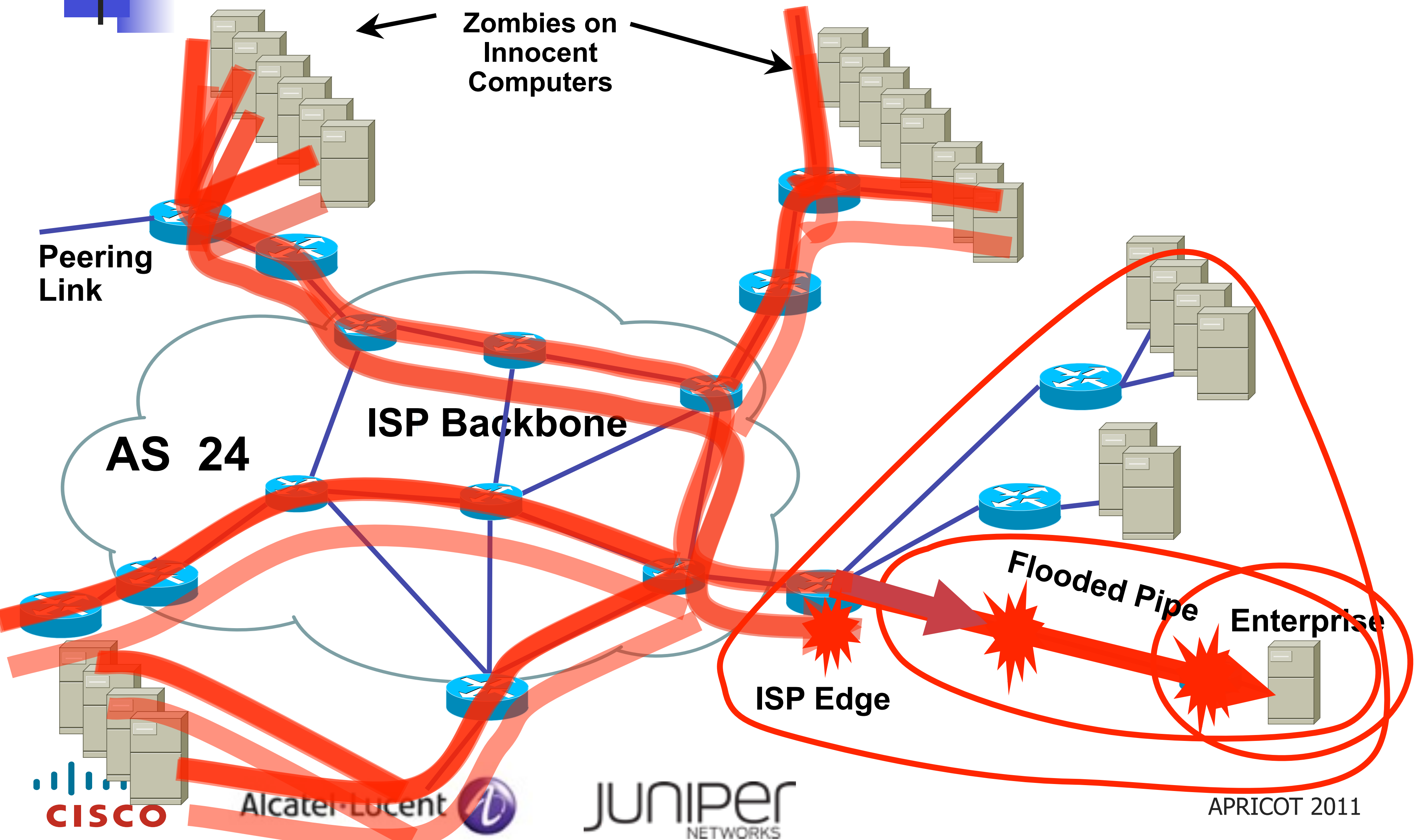
- Use valid protocols
- Spoof source IP
- Massively distributed
- Variety of attacks



DoS: The Procedure



An SP View: Denial of Service



Where to go to get more?

- **Operators Security Curriculum**
  - Sessions recorded over time which builds a library for all SPs to use for their individual training, staff empowerment, and industry improvements.
- http://www.nanog.org/ispsecurity.html
- http://www.apricot.net/apricot2005/workshop.html

Infrastructure Attacks

- Infrastructure attacks are increasing in both volume and sophistication
  - Sites with Cisco documents and presentations on routing protocols (and I don't mean Cisco.com)
  - Marked increase in presentations about routers, routing and Cisco IOS vulnerabilities at conferences like Blackhat, Defcon and Hivercon
  - Router attack tools and training are being published
- Why mount high-traffic DDOS attacks when you can take out your target's gateway routers?
- Hijacked routers are valuable in the spam world, which has a profit driver
- Router compromise (0wn3d) due to weak password

From Bad to Worms

- Worms have emerged as the new security reality
- Old worms never die!
  - Millions of UPnP and Slammer packets still captured daily
- Most worms are intended to compromise hosts
- Worm propagation is dependant on network availability
- Worms and DoS are closely related
  - Secondary worm effects can lead to denial of service
  - Worms enable DoS by compromising hosts → BOTnets
- Perimeters are crumbling under the worm onslaught (VPN/mobile workers, partners, etc.)

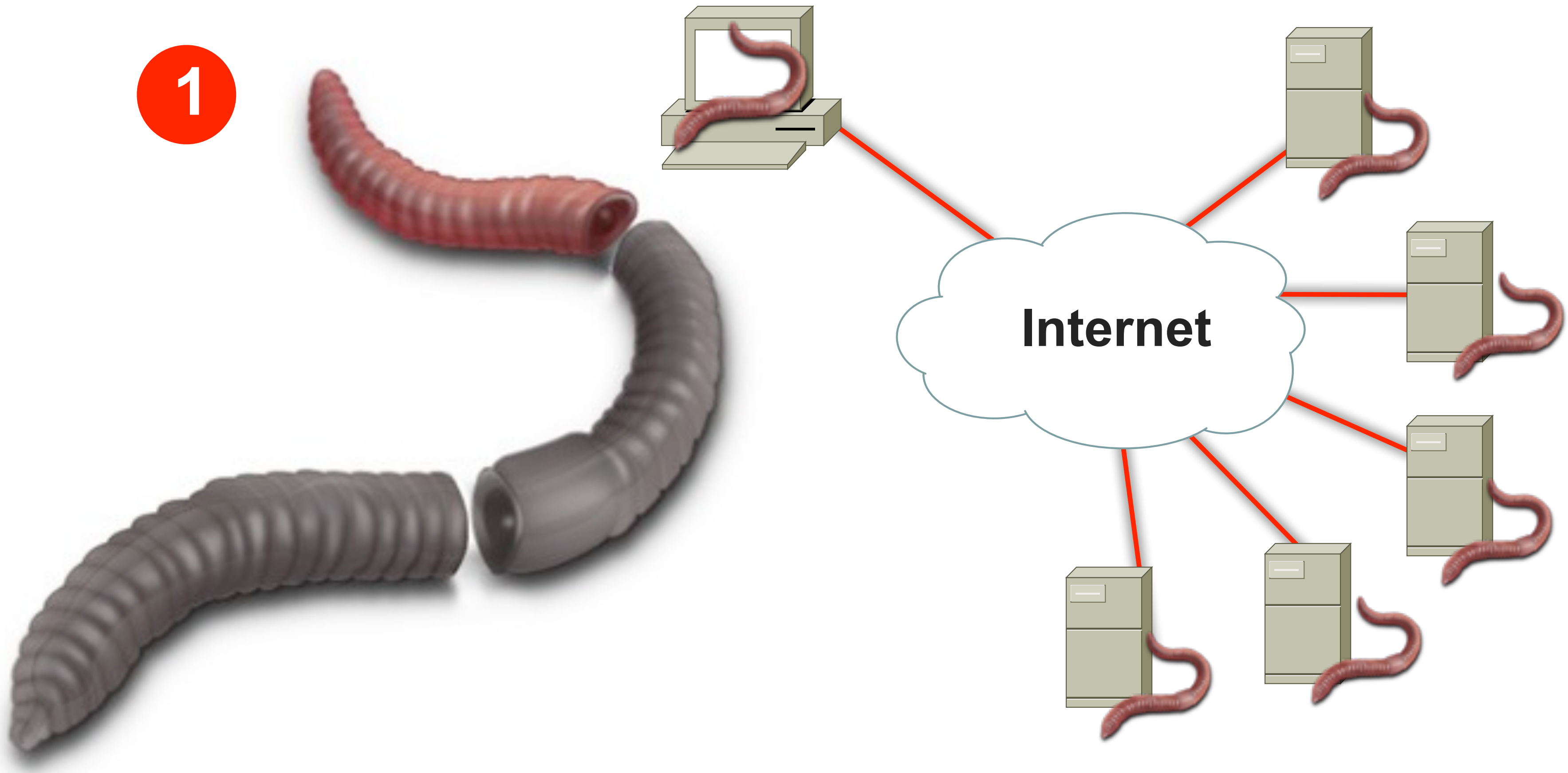
Anatomy of a Worm

1—The Enabling Vulnerability

2—Propagation Mechanism

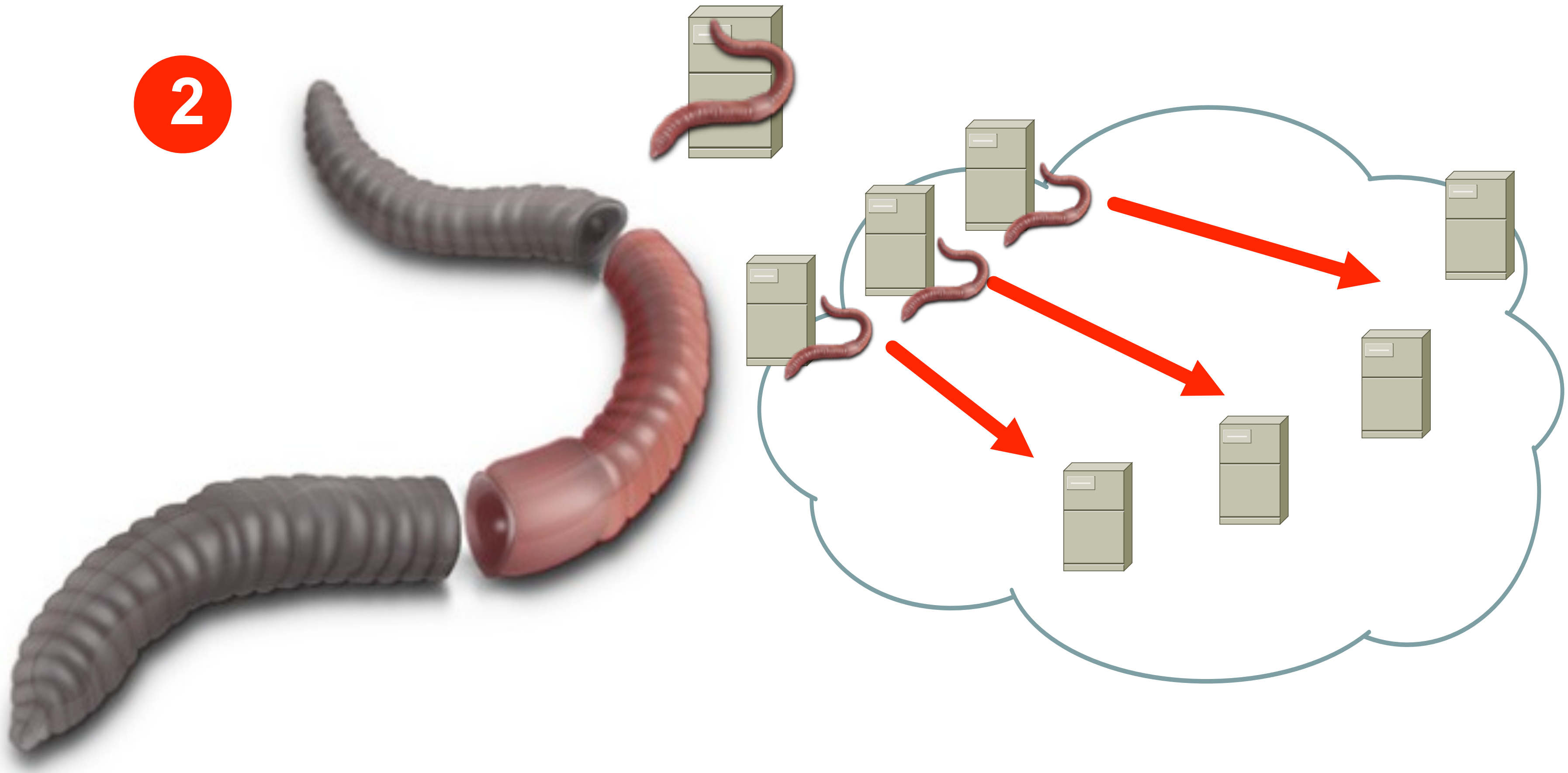
3—Payload

The Enabling Vulnerability



A Worm Installs Itself Using an Exploit Vector on a Vulnerable System

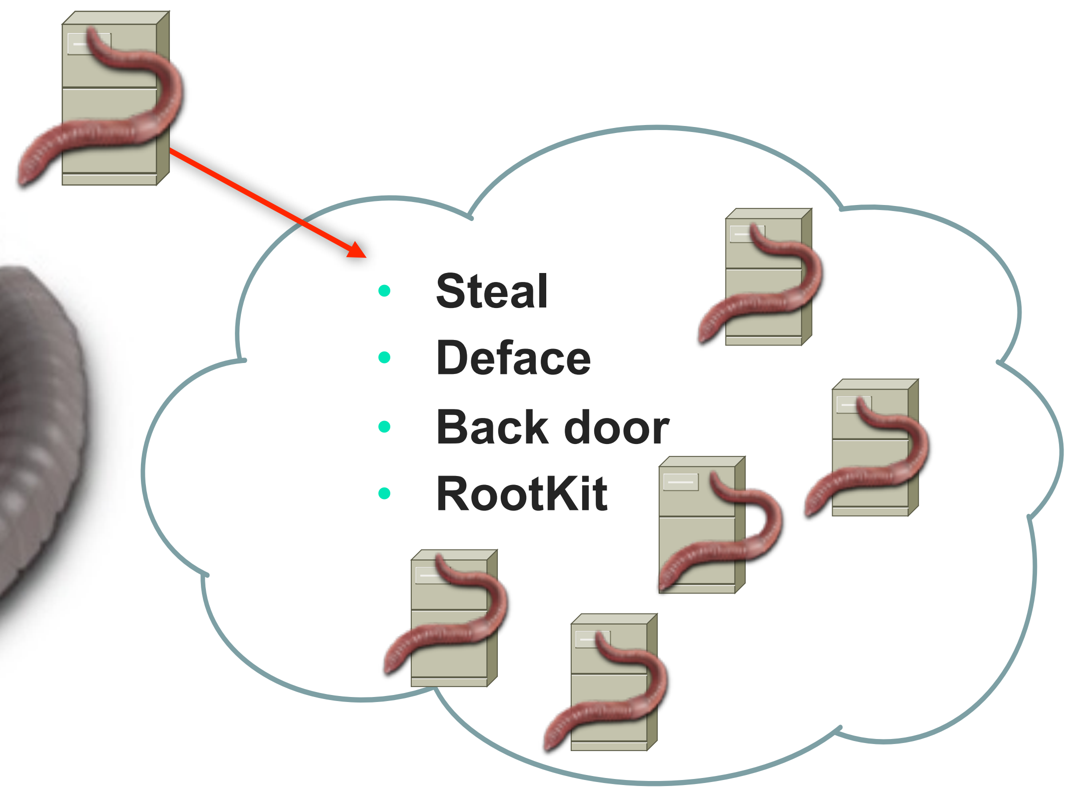
Propagation



After Gaining Access to Devices,
Worm Replicates and Selects New Targets

Payload

3



Worms and the Infrastructure

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

How Do You Respond?

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

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

Six Phases of Incident Response

PREPARATION
Prep the network
Create tools
Test tools
Prep procedures
Train team
Practice

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

CLASSIFICATION
What kind of attack is it?

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

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

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

Preparation

Preparation—Develop and Deploy a Solid Security Foundation

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

Preparation

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

Identification

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

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

Ways to Detect

- Customer call
  - "The Internet is down"
- Unexplained changes in network baseline
  - SNMP: line/CPU overload, drops
  - Bandwidth
  - NetFlow
- ACLs with logging
- Backscatter
- Packet capture
- Network IDS
- Anomaly detection

Network Baselines

- NMS baselines
- Unexplained changes in link utilization
  - Worms can generate a lot of traffic, sudden changes in link utilization can indicate a worm
- Unexplained changes in CPU utilization
  - Worm scans can affect routers/switches resulting in increased CPU both process and interrupt switched
- Unexplained syslog entries
- These are examples
  - Changes don't always indicate a security event!
  - Need to know what's normal in order to identify abnormal behavior

Classification

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

Classification

- Qualify and quantify the attack without jeopardizing services availability (e.g., crashing a router):
  - What type of attack has been identified?
  - What's the effect of the attack on the victim(s)?
  - What next steps are required (if any)?
- At the very least:
  - Source and destination address
  - Protocol information
  - Port information

Traceback

- Traceback—what are the sources of the attack?
  - How to trace to network ingress points
  - Your Internet connection is NOT the only vector
  - Understand your topology!
- Traceback to network perimeter
  - NetFlow
  - Backscatter
  - Packet accounting
- Retain attack data
  - Use to correlate interdomain traceback
  - Required for prosecution
  - Deters future attacks
  - Clarify billing and other disputes
  - Post mortem analysis

Reaction

Reaction—Do Something to Counter the Attack

- Should you mitigate the attack?
  - Where? How?
- No reaction is a valid form of reaction in certain circumstances
- Reaction often entails more than just throwing an ACL onto a router

Post Mortem

Post Mortem—Analyze the Event

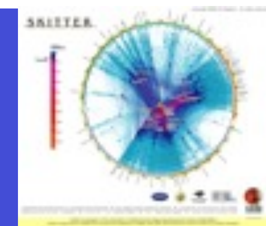
- The step everyone forgets!
- What worked? What didn't? How can we improve?
- What can be done to build build defense against repeat occurrences
- Was the DOS attack you just handled the real threat? Or was it a smoke screen for something else that just happened?
- What can you do to make it faster,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

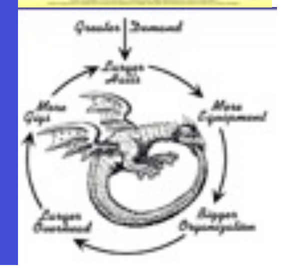
Resources, headcount, etc.

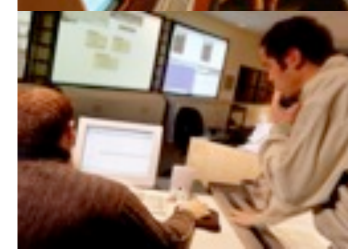
Security Workshop

OS Overview

Juniper OS Configuration

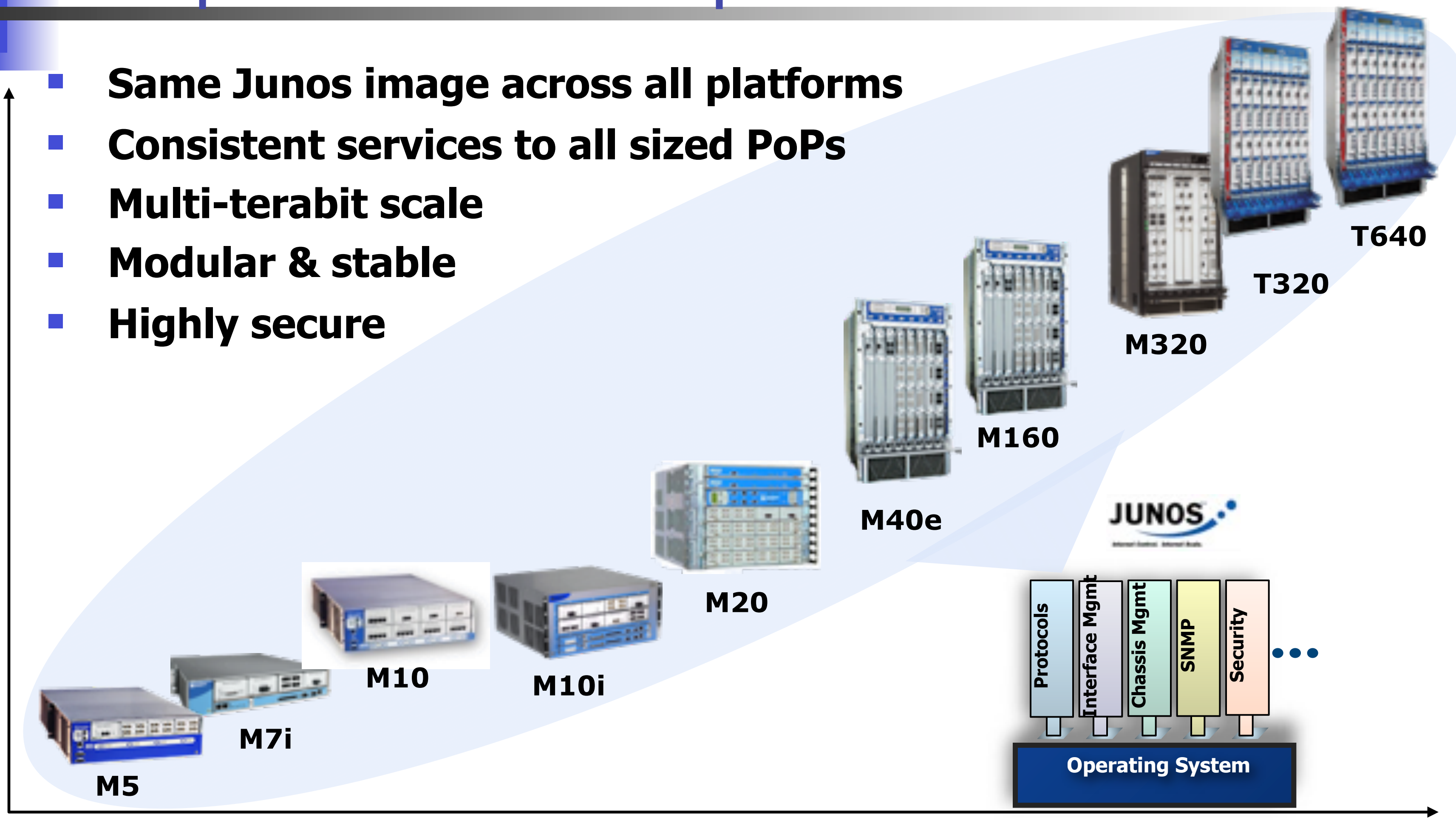






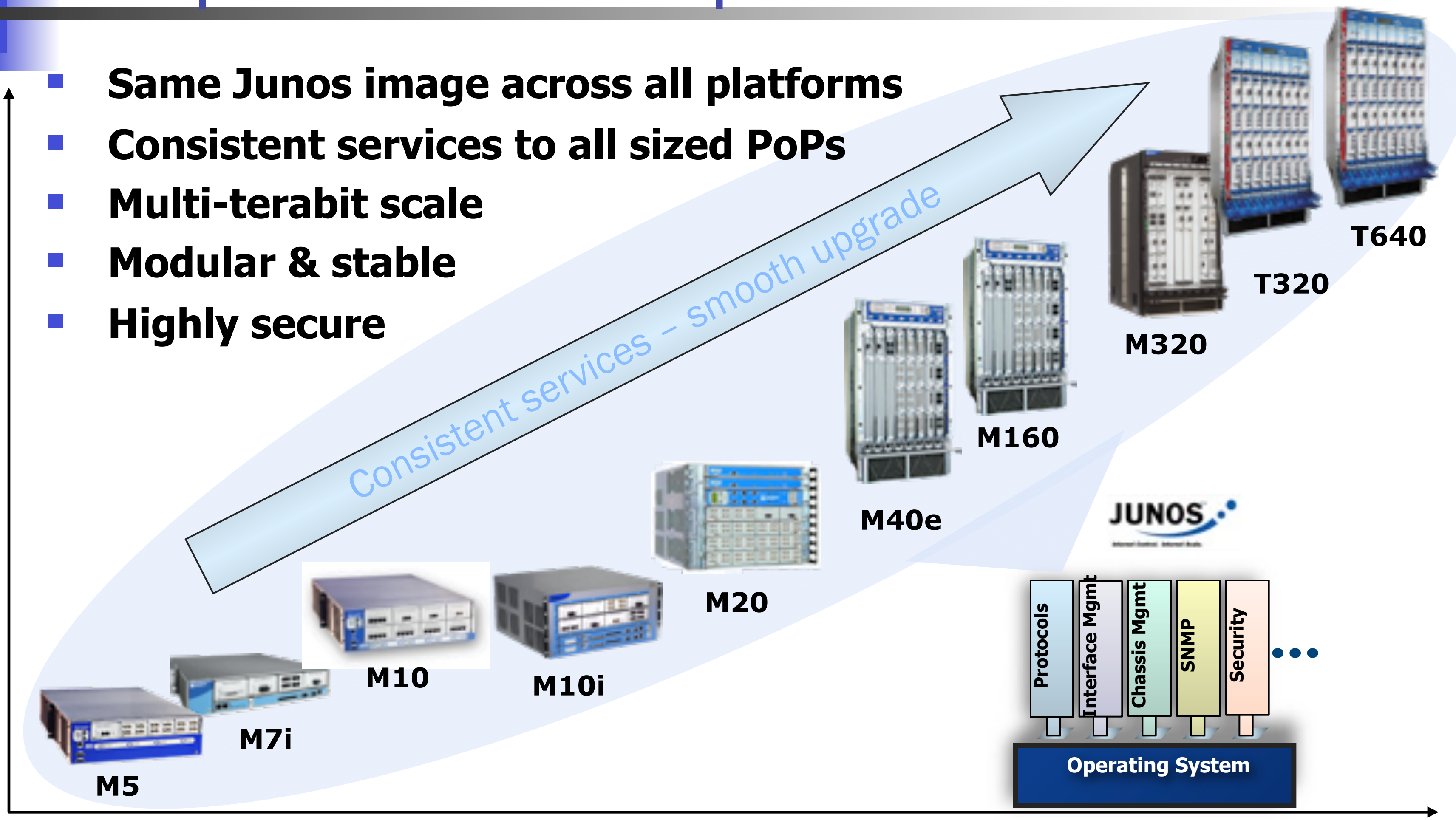
Juniper M-T series product line

- Same Junos image across all platforms
- Consistent services to all sized PoPs
- Multi-terabit scale
- Modular & stable
- Highly secure



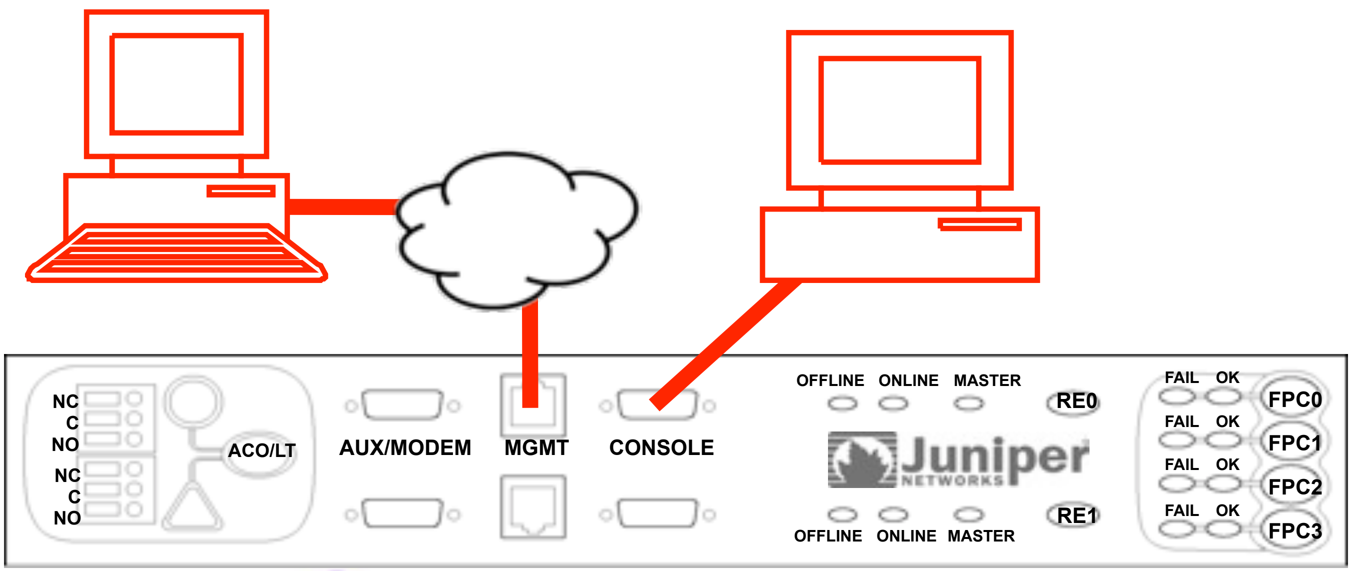
Juniper M-T series product line

- Same Junos image across all platforms
- Consistent services to all sized PoPs
- Multi-terabit scale
- Modular & stable
- Highly secure

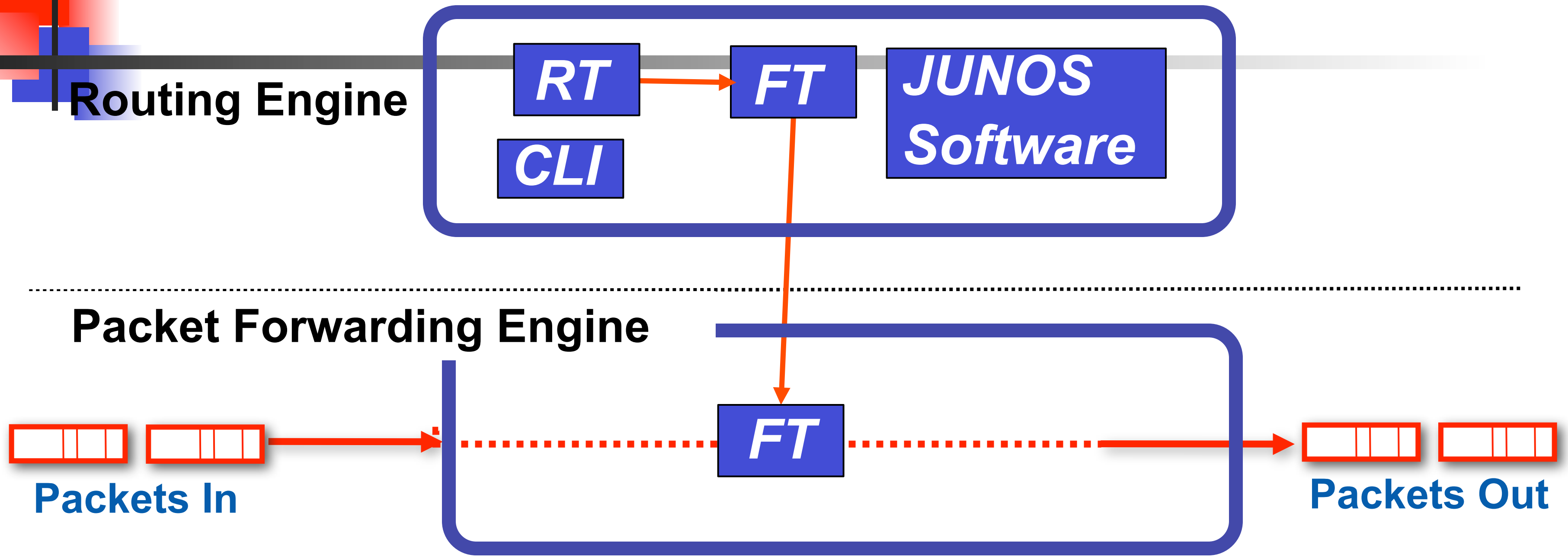


Access Router's Management Ports

- Console
  - Db9 EIA-232 @ 9600 Bps, 8/N/1 (preconfigured)
- Management port, using Telnet, SSH
  - Requires configuration

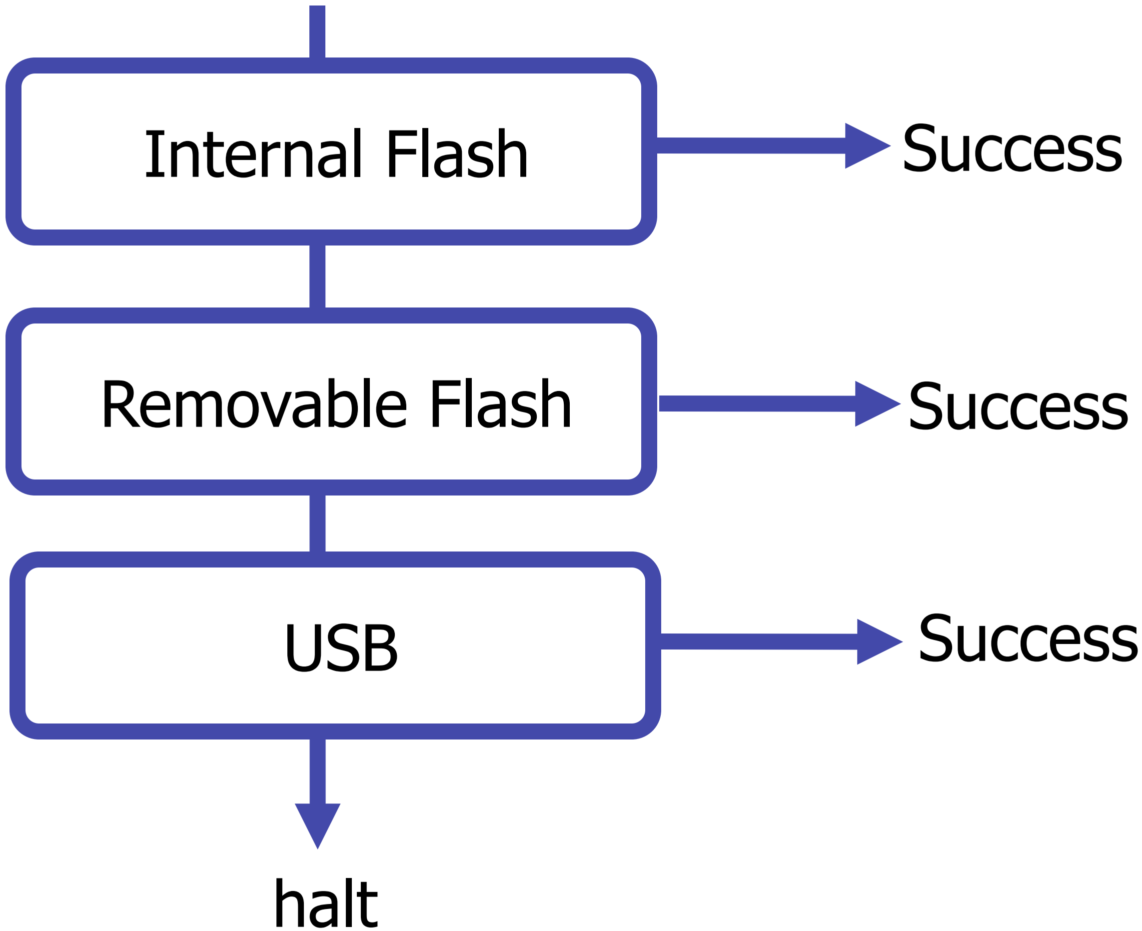


Separation of Control and Forwarding

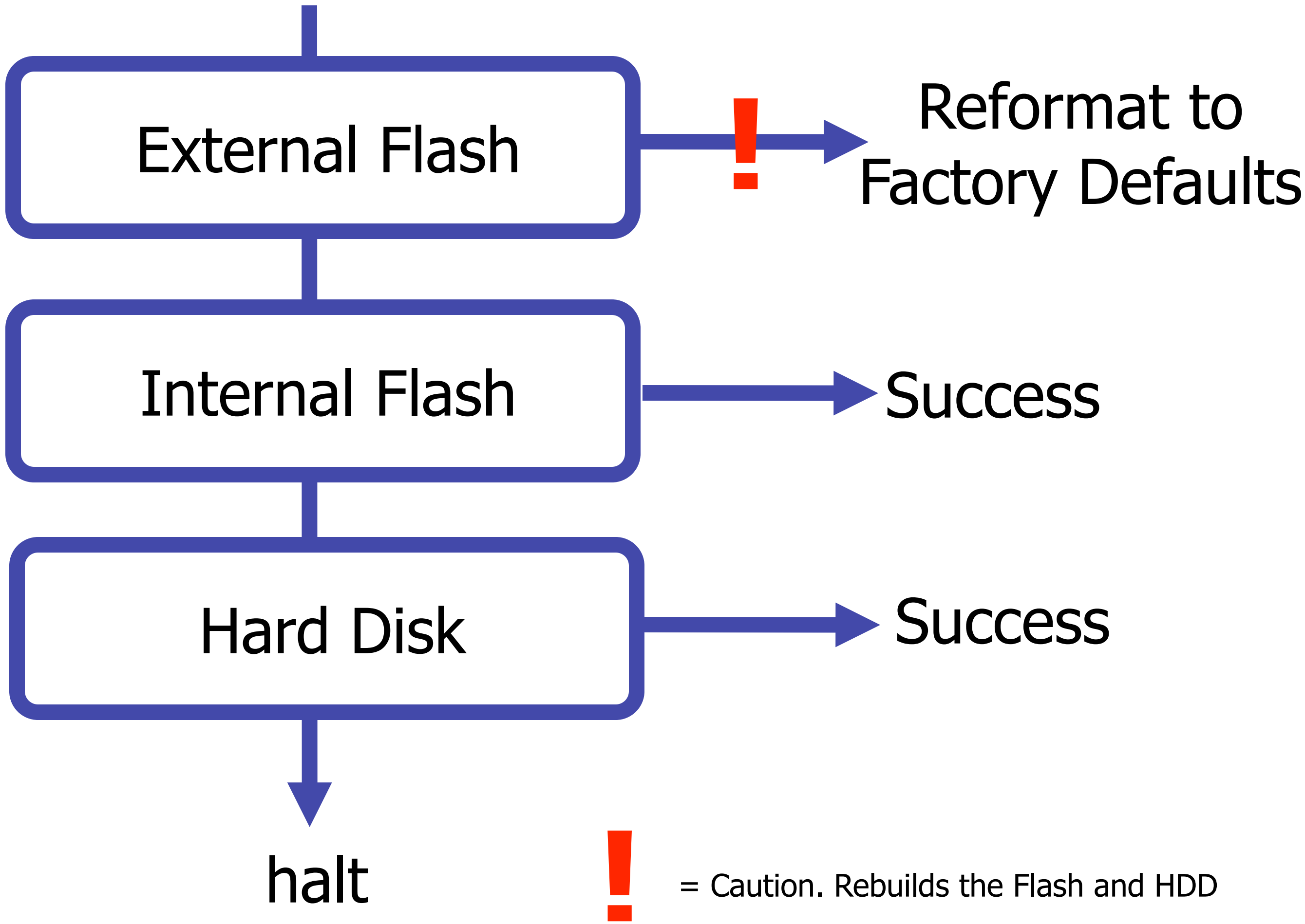


- J-series, M-series, and T-series platforms share a common design goal
  - Clean separation of control and forwarding
- Routing Engine maintains the routing table (RT) and a primary copy of the forwarding table (FT)
- Packet Forwarding Engine simply does what it is told, but it does it really fast

Boot Sequence J-Series



Boot Sequence J-Series



Initial Login – JUNOS

- Log in as root

```
. . .
starting local daemons:.
Fri Jan 17 22:23:32 UTC 1997

Amnesiac (ttyd0)

login: root
Last login: Fri Jan 17 22:21:55 on ttyd0

--- JUNOS 5.2R2.3 built 2002-03-23 02:44:36 UTC

Terminal type? [vt100] <enter>
root@%
```

Amnesiac indicates a factory default configuration

BSD shell prompt

Log In

- Router administrator configures login ID and password for each user
- Example session

```
lab2 (ttyd0)

login: perkins
Password:
Last login: Fri Feb 18 19:23:16 on ttyd0
Copyright (c) 1980, 1983, 1986, 1988, 1990, 1991, 1993, 1994
        The Regents of the University of California.
---JUNOS 4.0R1 built 2000-02-10 09:29:44 UTC

perkins@lab2>
```

CLI Modes and Feature Overview

- CLI operational mode
  - Editing command lines
  - Command completion/history
  - Context-sensitive and documentation-based help
  - UNIX-style pipes
- CLI configuration mode
  - Object-oriented hierarchy
  - Configuration groups
  - Jumping between levels
  - Candidate configuration with sanity checking
  - Automatic rollback capability
  - Showing portions of configuration while configuring
  - Running operational-mode commands from within configuration
  - Saving, loading, and deleting configuration files
  - Wildcard deletes

CLI Modes

- Operational mode
  - Monitor and troubleshoot the software, network connectivity, and router hardware

```
lab@host>
```

The > character identifies operational mode

- Configuration mode
  - Configure the router, including interfaces, general routing information, routing protocols, user access, and system hardware properties

```
[edit]
lab@host#
```

The # character identifies configuration mode

EMAC Style shortcuts

Start of line	Ctrl-a
End Of line	Ctrl-e
Delete line	Ctrl-u, Ctrl-x
Delete cursor to end of	Ctrl-k
Delete previous word	Ctrl-w
Redraw line	Ctrl-l
Search History	Ctrl-r
	Many more…

Command Completion

- Space bar completes a command

```
root@lab2> sh<space>ow i<space>
'i' is ambiguous.

Possible completions:
  igmp                 Show information about IGMP
  interfaces           Show interface information
  isis                 Show information about IS-
   IS

root@lab2> show i
```

- Tab key completes a variable

Context-Sensitive Help

Type a question mark (?) anywhere on command line

```
lab@host> ?
Possible completions:
  clear                    Clear information in the system
  configure                Manipulate software configuration information
  file                     Perform file operations
  help                     Provide help information
  . . .
lab@host> clear ?
Possible completions:
  arp                      Clear address resolution information
  bfd                      Clear Bidirectional Forwarding Detection
                                     information
  bgp                      Clear Border Gateway Protocol information
  cli                      Clear command-line interface settings
  firewall                 Clear firewall counters
```

Topical Help

The `help topic` command provides information on general concepts

```
lab@host> help topic icmp ?
Possible completions:
  address                    IP addresses to include in router advertisements
  lifetime                   How long addresses in advertisements are valid
  min-advertisement-interval  Time between router advertisements
  traceoptions               Trace options for ICMP
lab@host> help topic icmp lifetime

Modify the Router Advertisement Lifetime

   The lifetime field in router advertisement messages indicates how long
   a host should consider the advertised address to be valid. If this
   amount of time passes and the host has not received a router
   advertisement from the server, the route marks the advertised…….
```

Getting Help on Configuration Syntax

The `help reference` command provides configuration-related information

```
lab@host> help reference icmp lifetime
lifetime
      Syntax

   lifetime seconds;

      Hierarchy Level

   [edit protocols router-discovery interface interface-name]

      Description

   How long the addresses sent by the server in its router advertisement
   packets are valid. This time must be long enough so that another
   . . . .

      Options

   seconds--Lifetime value. A value of 0 indicates that one or more
   addresses are no longer valid.
   Range: 0, max-advertisement-interval value through 2 hours, 30
   minutes (9000 seconds), specified in seconds
   Default: 1800 seconds (30 minutes; three times the default
```

By the way…apropos

Where is keyword stub?

```
[edit]
lab@Hong_Kong_01# help apropos stub
…
[edited for brevity]
…
set protocols ospf sham-link no-advertise-local
set protocols ospf area <area_id>
set protocols ospf area <area_id> stub
set protocols ospf area <area_id> stub default-metric
   <default-metric>
set protocols ospf area <area_id> stub summaries
set protocols ospf area <area_id> nssa
```

Using | (Pipe)

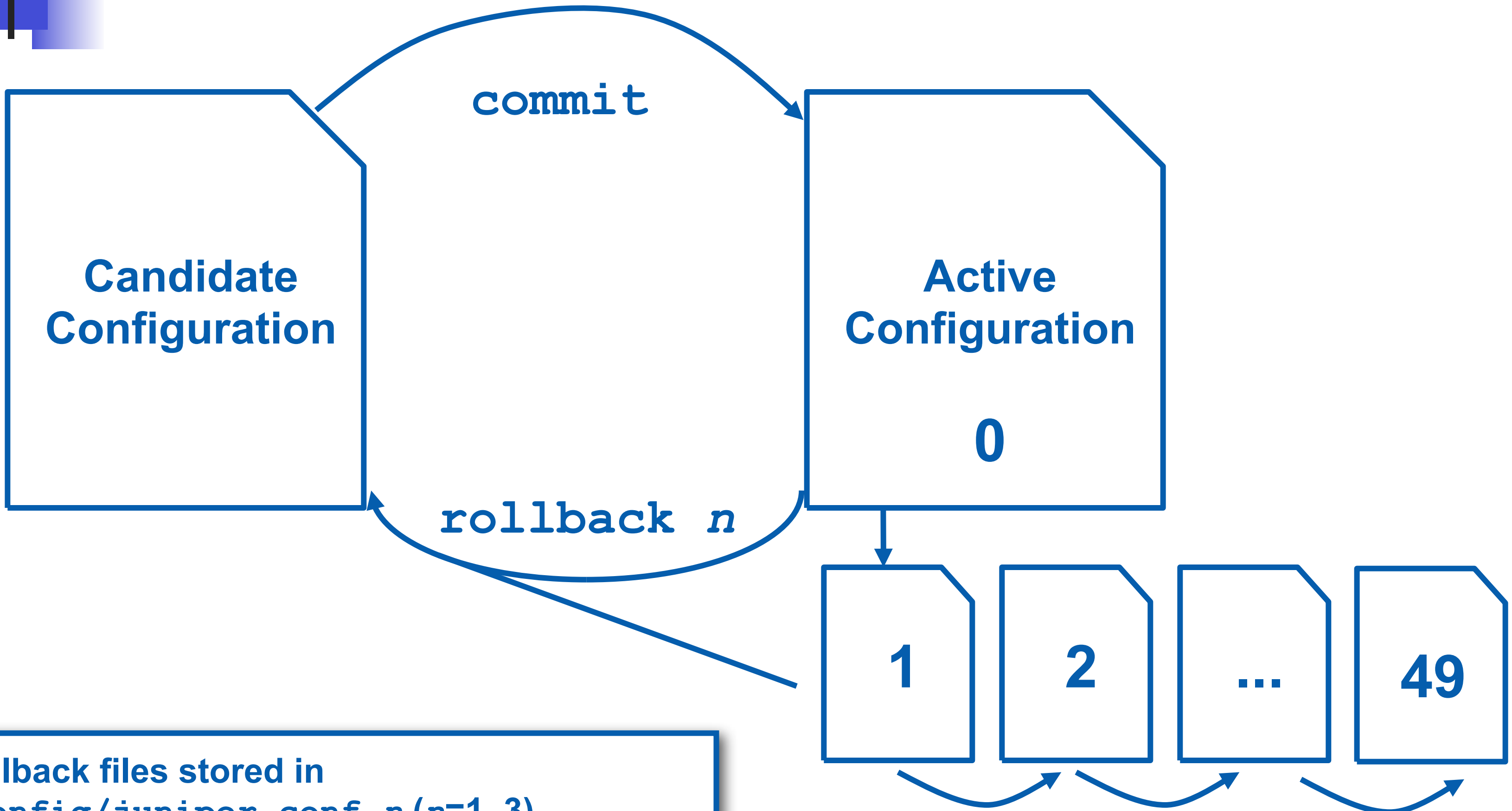
- The pipe function is used to filter output
  - Available in all modes and context

Using | (Pipe)

- The pipe function is used to filter output
  - Available in all modes and context

```
user@host> show route | ?
Possible completions:
  count                    Count occurrences
  display                  Display additional information
  except                   Show only text that does not match a pattern
  find                     Search for the first occurrence of a pattern
  hold                     Hold text without exiting the --More-- prompt
  last                     Display the last screen of lines in the output
  match                    Show only text that matches a pattern
  no-more                  Don't paginate output
  request                  Make system-level requests
  resolve                  Resolve IP addresses
  save                     Save output text to a file
  trim                     Trim specified number of columns from start of line
```

Activating a Configuration (1 of 2)



Rollback files stored in
`/config/juniper.conf.`*n* (*n*=1–3)
`/var/db/config/juniper.conf.`*n* (*n*=4–49)

Entering Configuration Mode

- Type `configure` or `edit` at the CLI operational-mode prompt

```
root@lab2> configure
Entering configuration mode
[edit]
root@lab2#
```

- To allow a single user to edit the configuration, type `configure exclusive`
- `configure private` allows the user to edit a private copy of the candidate configuration
  - Multiple users can edit private candidate configurations simultaneously
  - At commit time, the user's private changes are merged back

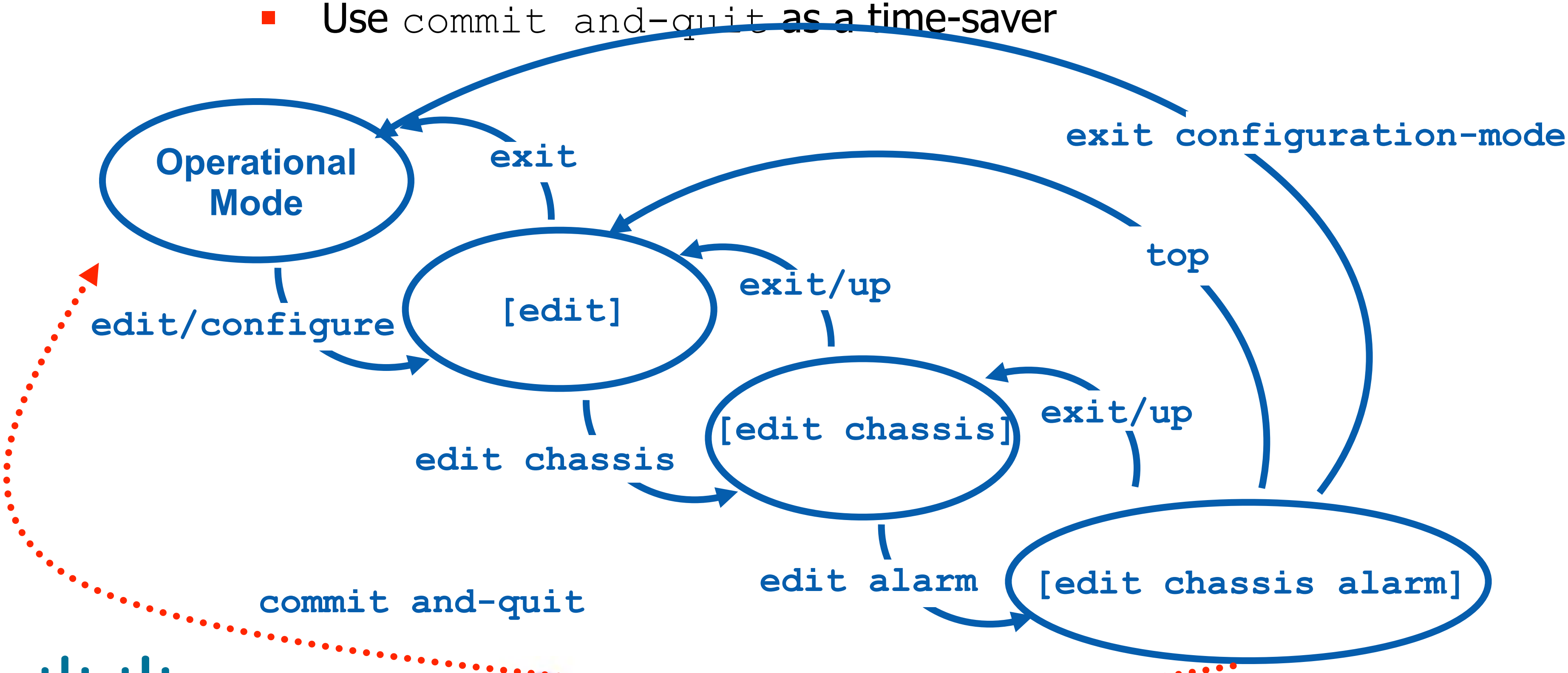
Activating a Configuration … Commit

- Commit
  - Use `commit confirmed` to temporarily activate a configuration (default is 10 minutes).If configuration is not confirmed, router returns to previous configuration automatically; a second `commit` confirms the changes
- Use the `synchronize` switch to mirror the new configuration to a backup RE
- Support for scheduled and commented commits
  - Use the `commit at` *`time`* option (Release 5.5)

```
[edit]
user@host# commit at 20:01:00
configuration check succeeds
commit at will be executed at 2003-08-08 20:01:00 UTC
The configuration has been changed but not committed
Exiting configuration mode
```

Exiting Configuration Mode

- Exiting levels
  - Use `exit` from top level
  - Use `exit configuration-mode` from any level
  - Use `commit and-quit` as a time-saver



Configuration Hierarchy

- Create a hierarchy of configuration statements
  - Enter commands in CLI configuration mode

```
root@lab2# set chassis alarm sonet lol red
```

  - And the resulting configuration hierarchy is created…

```
chassis {
   alarm {
        sonet {
           lol red;
        }
   }
}
```

  - Delete commands

```
root@lab2# delete chassis alarm sonet lol
```

Configuring Logical Interfaces

- Use the `set` command to configure a logical interface using the unit number
  - For example:

```
lab@omaha> configure
[edit]
lab@omaha# set interfaces so-1/0/3 unit 40 dlci 40
```

- Or park yourself at the `unit` level:

```
lab@omaha> configure
[edit]
lab@omaha# edit interfaces so-1/0/3 unit 40
[edit interfaces so-1/0/3 unit 40]
lab@omaha# set dlci 40
```

Viewing Candidate Configuration

You can display just the portions that concern you from the root of the hierarchy…

```
[edit]
 user@host# show chassis alarm
 sonet {
          los red;
          pll yellow;
      }
 [edit]


 user@host# edit chassis alarm
 [edit chassis alarm]
 user@host# show
 sonet {
          los red;
          pll yellow;
      }
 [edit chassis alarm]
```

…or use `edit` to park yourself at a specific sub-hierarchy

`run` / do is Cool

- Use the run command to execute operational-mode CLI commands from within configuration
  - Can be a real time-saver when testing the effect of a recent change

```
[edit interfaces so-0/1/1]
lab@Amsterdam# set unit 0 family inet address 10.0.24.2/24


[edit interfaces so-0/1/1]
lab@Amsterdam# commit
commit complete


[edit interfaces so-0/1/1]
lab@Amsterdam# run ping 10.0.24.1 count 1
PING 10.0.24.1 (10.0.24.1): 56 data bytes
64 bytes from 10.0.24.1: icmp_seq=0 ttl=255 time=0.967 ms

--- 10.0.24.1 ping statistics ---
1 packets transmitted, 1 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.967/0.967/0.967/0.000 ms
```

Test configuration changes without leaving configuration mode with `run`

Initial Configuration Checklist

- The following items are normally configured at initial system installation:
  - Root password
  - Host name
  - Domain name and DNS server address
  - Configuration file compression
  - System logging
  - Out-of-band management interface
  - Default and backup routers for management network
  - Configure system services for remote access
  - User accounts
  - System time
  - Loopback and transient interfaces
  - Remaining configuration needed to place the router into service (protocols,

Initial Configuration (1 of 10)

- Log in as root

Amnesiac indicates a factory default configuration

```
. . .
starting local daemons:.
Fri Jan 17 22:23:32 UTC 1997


Amnesiac (ttyd0)


login: root
Last login: Fri Jan 17 22:21:55 on ttyd0


--- JUNOS 5.2R2.3 built 2002-03-23 02:44:36 UTC


Terminal type? [vt100] <enter>
root@%
```

BSD shell prompt

Initial Configuration (2 of 10)

- Enter configuration mode

  root> configure

  [edit]

  root#

- Configure root password
  - Plain text

    root# set system root-authentication plain-text-password

  - Pre-encrypted password

    root# set system root-authentication encrypted-password *encrypted-password*

Do *not* enter a clear text password in this mode!

Initial Configuration - IOS(2a of 10)

- Enter configuration mode

```
Root# configure
Root(config)#
```

- Configure enable password
  - Plain text

```
Root(config)# enable password  password
```

  - Pre-encrypted password

```
Root(config)# enable secret 5 $1!Q$hjHJHGJGJHGY
```

Do *not* enter a clear text password in this mode!

Initial Configuration (3 of 10)

- Configure router name

```
[edit]
root# set system host-name lab2
```

- Configure router domain name

```
[edit]
root# set system domain-name domain-name.tld
```

- Configure name server address

```
[edit]
root@# set system name-server ns-address
```

Initial Configuration – IOS (3a of 10)

- Configure router name

  `hostname` *lab2*

- Configure router domain name

  `ip domain-name` *domain-name.tld*

- Configure name server address

  [edit]

  `ip name-server` *ns-address*

Initial Configuration (4 of 10)

- Adjust syslog parameters as needed
  - Interactive command and configuration change logging is a good idea
  - Adjusting archive settings for more history also recommended

```
[edit system syslog]
root@lab2# show
user * {
    any emergency;
}
file messages {
    any notice;
    authorization info;
    archive size 1m files 20;
}
file cli-commands {
    interactive-commands any;
    archive size 1m files 10;
}
file config-changes {
    change-log info;
    archive size 1m files 10;
}
```

Archive settings adjusted on default syslog file

Interactive commands and configuration changes

Initial Configuration (5 of 10)

- Commit changes so far

```
[edit]
root# commit
commit complete

[edit]
root@lab2#
```

Note host name takes effect after the commit

- Configure management interface IP address and prefix

```
[edit]
root@lab2# set interfaces fxp0 unit 0 family inet address ip-address/
   prefix-length
```

- Define a backup router
  - Used when routing daemon is not running
    - Required when using redundant Routing Engines

```
[edit]
root@lab2# set system backup-router gateway-address
```

Initial Configuration (6 of 10)

- Define static route for OoB management network

```
[edit]
root@lab2# edit routing-options


[edit routing-options]
root@lab2# set static route ip-address/prefix-length
 next-hop OoB-next-hop-address no-readvertise
```

- Configure system services for remote access

```
[edit]
root@lab2# set system services ssh
[edit]
root@lab2# set system services telnet
[edit]
root@lab2# set system services ftp
```

Initial Configuration – IOS (6a of 10)

- Define static route for OoB management network
- Static route

```
Lab2(config)# ip route destination mask next-hop ???
```

- Configure system services for remote access

```
line vty 0 4
login
password cisco
```

More on Banner

- Legal requirements may vary, but be explicit

"This system may be accessed by authorized persons only. Unauthorized access is forbidden and subject to criminal and civil penalties, as well as company disciplinary actions. By accessing this system you acknowledge that your actions will be monitored."

Initial Configuration (7 of 10)

- Configure user accounts
  - Use predefined login classes, or create your own

```
[edit system login]
root@lab2# show
user dr-data {
    full-name "The Doctor 'O Data";
    uid 2003;
    class superuser;
    authentication {
        encrypted-password "$1$B78jkPLd$8VVjFv6D.ZQQev/5rstET0"; # SECRET-DATA
    }
}
```

The user ID is created automatically when not explicitly configured

The commands used to create the *`dr-data`* user account, courtesy of `display set`

```
[edit system login]
root@lab2# show | display set
set system login user dr-data full-name "The Doctor 'O Data"
set system login user dr-data uid 2003
set system login user dr-data class superuser
set system login user dr-data authentication encrypted-password "$1$B78jkPLd
$8VVjFv6D.ZQQev/5rstET0"
```

More on classes

- Define expected roles
  - One class per role
- Super-user
  - Use sparingly, even for yourself, as accidents happen when wielding enormous power!!! ☺
- View-only
  - Good for script engines
  - Limited NOC capability
- Newer OS's allow enormous granularity
  - Interface control vs. control plane (OSPF BGP etc)
- Commit scripts

Initial Configuration IOS (7a of 10)

- Configure user accounts
  - Use predefined privilege, or create your own

```
username lab privilege 15 password 7 09404F0B485744
```

Initial Configuration (8 of 10)

- Configure time zone and manually set the time of day
  - Configure time zone:

```
[edit]
root@lab2# set system time-zone America/Los_Angeles
```

  - Set date and time manually

```
root@lab2> set date ?
Possible completions:
  <time>                   New date and time (YYYYMMDDhhmm.ss)
  ntp                      Set date/time using Network Time
   Protocol servers
root@lab2> set date 200405141017.20
Fri May 14 10:17:20 PDT 2004
```

- Or, configure NTP

Initial Configuration- IOS(8a of 10)

- Configure time zone and manually set the time of day
  - Configure time zone:

```
clock timezone _timezone
```

  - Set date and time manually

```
clock set hh:mm:ss
```

- Or, configure NTP

Initial Configuration (9 of 10)

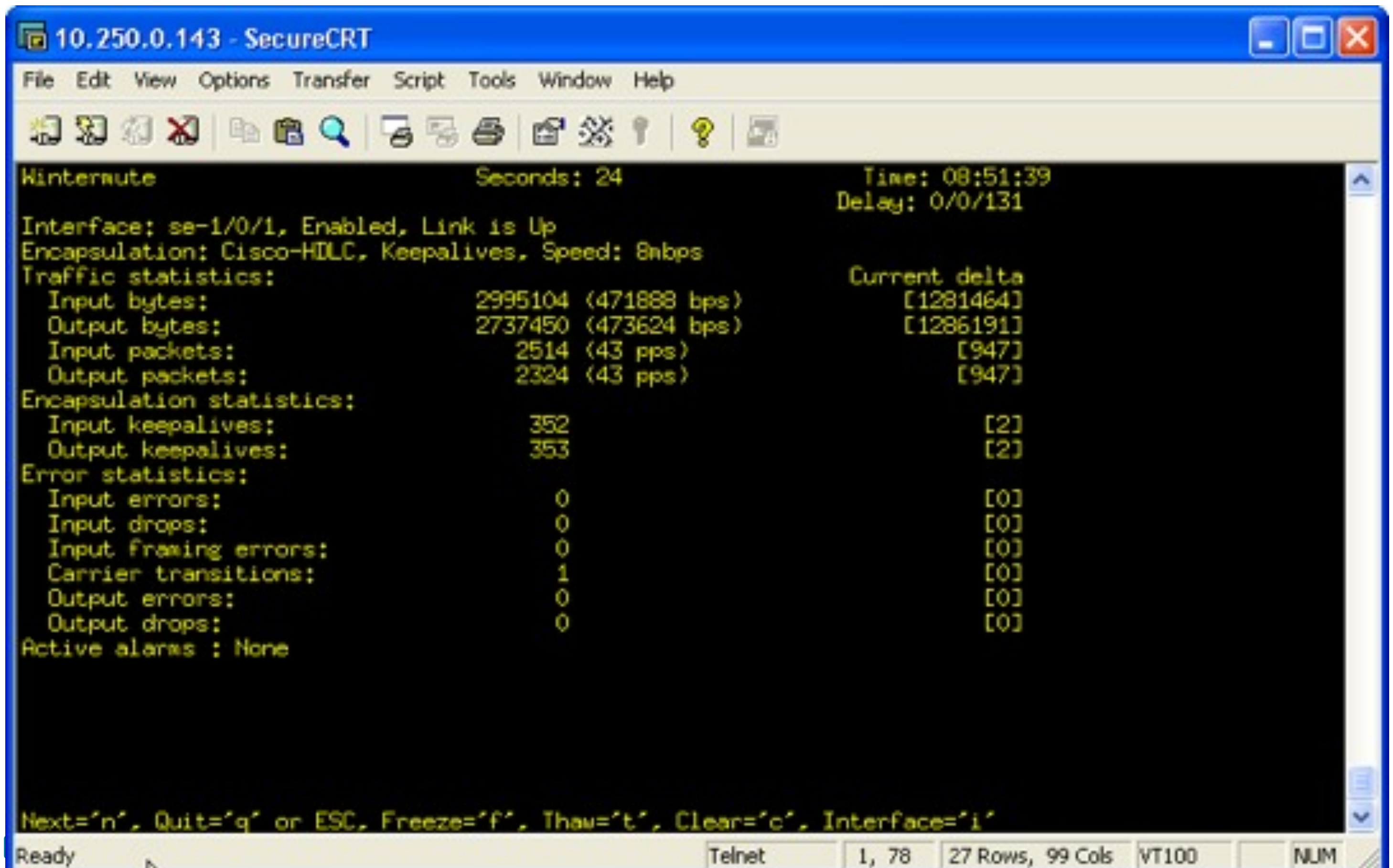
Configure loopback and transient interfaces

```
[edit interfaces]
root@lab2# set lo0 unit 0 family inet address 192.168.12.1

[edit interfaces]
root@lab2# set fe-0/0/2 unit 0 family inet address 10.0.13.2/24

[edit interfaces]
root@lab2# show lo0
unit 0 {
    family inet {
        address 192.168.12.1/32;
    }
}

[edit interfaces]
root@lab2# show fe-0/0/2
unit 0 {
    family inet {
        address 10.0.13.2/24;
    }
```

Loopback interface must use a /32

Monitoring an Interface

Use the `monitor interface` command for real-time statistics and error reports



Disabling, Deactivate, and Bounce

- Configuration mode deactivate and disable
  - `deactivate` causes the statement or hierarchy to be ignored
    - Comments out that portion of the configuration
  - `disable` administratively disables an interface or logical unit while retaining configured properties
- Use the operational-mode `request chassis` command to bounce FPC (PICs)
  - A warm boot of the FPC/PIC can clear problems
    - Less drastic then a chassis reboot and does not require configuration privileges

```
lab@Wintermute> request chassis fpc ?
Possible completions:
  offline                   Take FPC offline
  online                    Bring FPC online
  restart                   Restart FPC
  slot                      FPC slot number (0..6)
lab@Wintermute> request chassis fpc restart slot 2
Restart initiated, use "show chassis fpc" to verify
```

Network Utilities

- Extended PING capabilities all on one line

```
lab@HongKong> ping ?
       Possible completions:
         <host>                 Hostname or IP address of remote host
         atm                    Ping remote Asynchronous Transfer Mode node
         bypass-routing         Bypass routing table, use specified interface
         count                  Number of ping requests to send (1..2000000000 packets)
         detail                 Display incoming interface of received packet
         do-not-fragment        Don't fragment echo request packets (IPv4)
         inet                   Force ping to IPv4 destination
         inet6                  Force ping to IPv6 destination
         interface              Source interface (multicast, all-ones, unrouted packets)
         interval               Delay between ping requests (seconds)
         logical-router         Name of logical router
       + loose-source           Intermediate loose source route entry (IPv4)
         mpls                   Ping label-switched path
         no-resolve             Don't attempt to print addresses symbolically
         pattern                Hexadecimal fill pattern
         rapid                  Send requests rapidly (default coun
         record-route           Record and report packet's path (IP
         [deleted for brevity]
```

Access command options by clicking Advanced options

Network Utilities (Cont.)

- The `monitor traffic` command provides CLI access to the tcpdump utility
  - Only displays traffic originating or terminating on local RE
    - The best way to perform analysis of Layer 2 protocols in JUNOS software
    - Protocol filtering currently requires writing and reading from a file (hidden `write-file` and `read-file` options)

ICMP echo traffic

```
lab@host> monitor traffic interface ge-0/3/0 detail
Listening on ge-0/3/0, capture size 96 bytes
16:20:24.043006  In IP (tos 0x0, ttl 255, id 53152, offset 0, flags [none],
    length: 84) 10.0.16.1 > 10.0.16.2: icmp 64: echo request
16:20:24.043061 Out IP (tos 0x0, ttl 255, id 57238, offset 0, flags [none],
    length: 84) 10.0.16.2 > 10.0.16.1: icmp 64: echo reply
```

Tracing Overview

- Tracing is the JUNOS software equivalent of debug
  - Can be enabled on a production network
  - Requires configuration
  - Multiple options (flags) can be traced to a single file
- Generic tracing configuration syntax

```
[edit protocols protocol-name]
user@host# show
     traceoptions {
         file filename [replace] [size size] [files number] [no-
   stamp];
         flag flag [flag-modifier] [disable];
      }
```

The protocol/function being traced

Where to write the trace results

Flags identify what aspects of the protocol is traced and at what level of detail

Protocol Tracing

- Include the `traceoptions` statement at the `[edit protocols` *`protocol-name`*`]` hierarchy
  - Useful when troubleshooting configuration and interoperability problems
- A typical BGP tracing configuration is shown along with sample output:

```
[edit protocols bgp]
lab@host# show
traceoptions {
    file bgp-trace;
    flag open detail;
    flag update detail;
    flag keepalive detail;
}
lab@host> show log bgp-trace
. . .
Feb 19 16:07:47 BGP RECV 192.168.2.1+2705 -> 192.168.0.1+179
Feb 19 16:07:47 BGP RECV message type 1 (Open) length 45
Feb 19 16:07:47 BGP RECV version 4 as 10 holdtime 90 id 192.168.2.1 parmlen 16
Feb 19 16:07:47 BGP RECV MP capability AFI=1, SAFI=1
```

Analyzing Log and Trace Files

- Use the `show log` *`file-name`* CLI command to display contents of log and tracefiles
  - Hint: Get help on available options at the `more` prompt by entering an `h`
- Do not forget the CLI's pipe functionality; it makes log parsing a breeze!
  - Cascade pipe instances to evoke a logical AND search; use quotes to evoke a logical OR, as shown:

```
lab@host> show log messages | match fail

Jan 29 12:40:47  Montreal-3 rpd[2228]: RPD_ISIS_ADJDOWN: IS-IS lost
L2 adjacency to Amsterdam-3 on so-0/3/1.0, reason: 3-Way Handshake
Failed

lab@London> show log messages | match "fpc | error | kernel | panic"
```

Miscellaneous Log File Commands

- Monitor a log/trace in real time with the CLI's `monitor` command

```
user@host> monitor start filename
```

  - Shows updates to monitored file(s) until canceled, with piped output matching!
  - Use Esc-Q to enable/disable real-time output to screen
  - Issue `monitor stop` to cease all monitoring
- Log/trace file manipulation
  - Use the `clear` command to truncate (clear) log/trace files

```
user@host> clear log filename
```

  - Use the `file delete` command to delete log/trace files

```
user@host> file delete filename
```

Config Manipulation

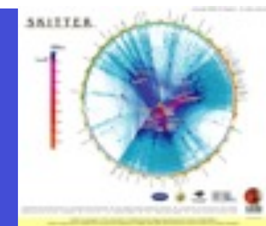
- Save a config to a file

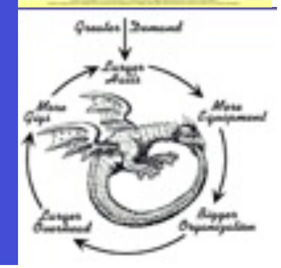
```
show conf | save myconfig

show conf | save ftp://lab:passwd@10.150.0.251

 archival {
        configuration {
            transfer-interval 1440;
            archive-sites {
                "ftp://user:password@10.10.10.1";
            }
        }
    }
```

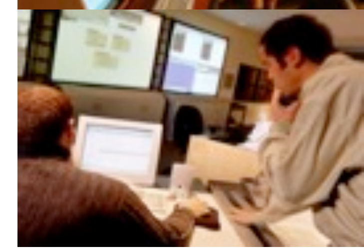
IOS <-> JUNOS

Basic CLI and Systems Management Commands	clock set	set date
	ping	ping
	reload	request system reboot
	send	request message
	show clock	show system uptime
	show environment	show chassis environment
	show history	show cli history
	show ip traffic	show system statistics
	show logging	show log show log file name
	Show processes	show system processes
	show running config	show configuration
	show tech-support	request support information
	show users	show system users
	show version	show version show chassis hardware
	terminal length	set cli screen-length
	terminal width	set cli screen-width
	trace	traceroute

Cisco IOS Configuration







Router Components

- Bootstrap – stored in ROM microcode – brings router up during initialisation, boots router and loads the IOS.
- POST – Power On Self Test - stored in ROM microcode – checks for basic functionality of router hardware and determines which interfaces are present
- ROM Monitor – stored in ROM microcode – used for manufacturing, testing and troubleshooting
- Mini-IOS – a.k.a RXBOOT/boot loader by Cisco – small IOS ROM used to bring up an interface and load a Cisco IOS into flash memory from a TFTP server; can also do a few other maintenance operations

Router Components

- RAM – holds packet buffers, ARP cache, routing table, software and data structure that allows the router to function; running-config is stored in RAM, as well as the decompressed IOS in later router models
- ROM – starts and maintains the router
- Flash memory – holds the IOS; is not erased when the router is reloaded; is an EEPROM [Electrically Erasable Programmable Read-Only Memory] created by Intel, that can be erased and reprogrammed repeatedly through an application of higher than normal electric voltage
- NVRAM – Non-Volatile RAM - holds router configuration; is not erased when router is reloaded

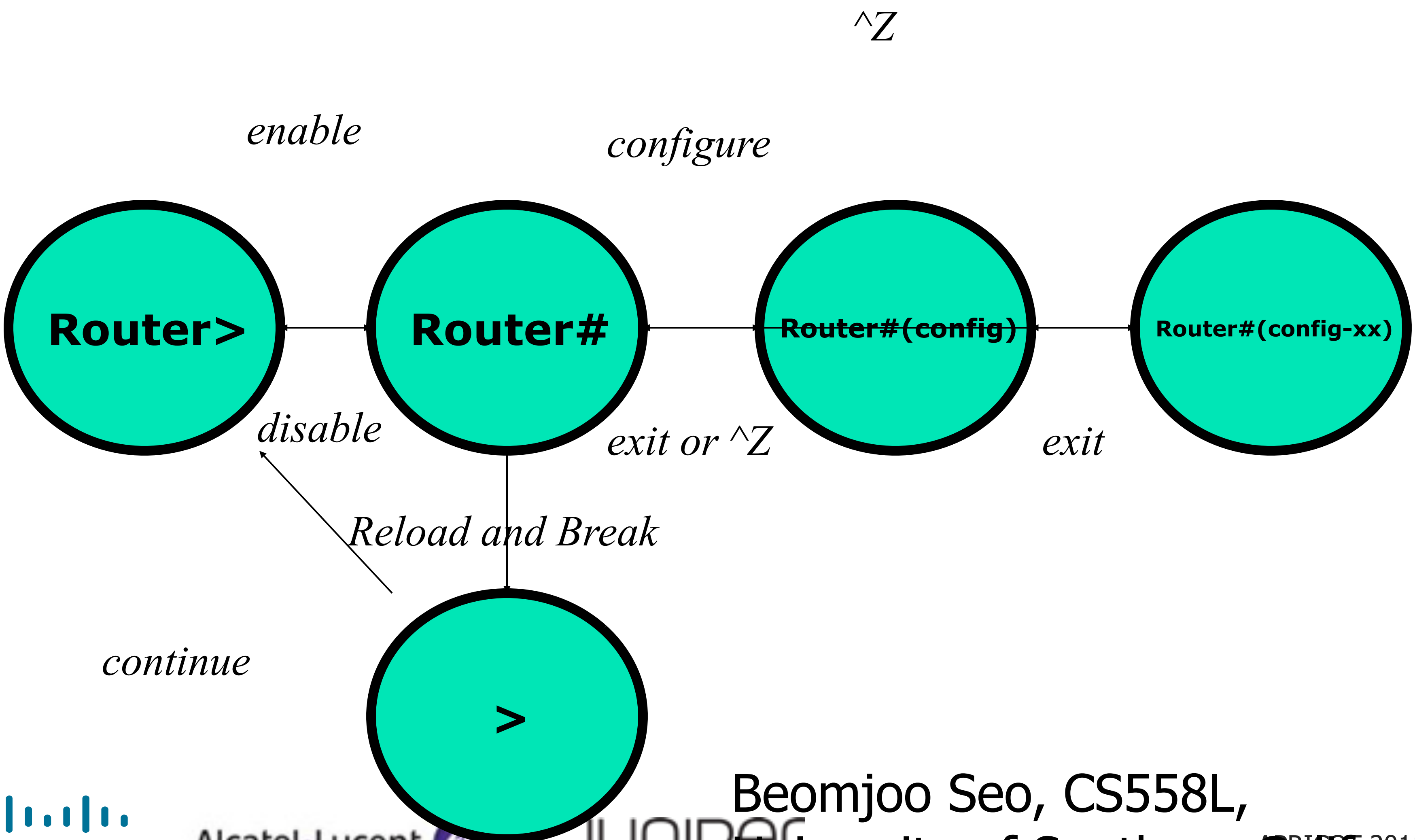
Router Components

- Config-Register – controls how router boots; value can be seen with "`show version`" command; is typically 0x2102, which tells the router to load the IOS from flash memory and the `startup-config` file from NVRAM

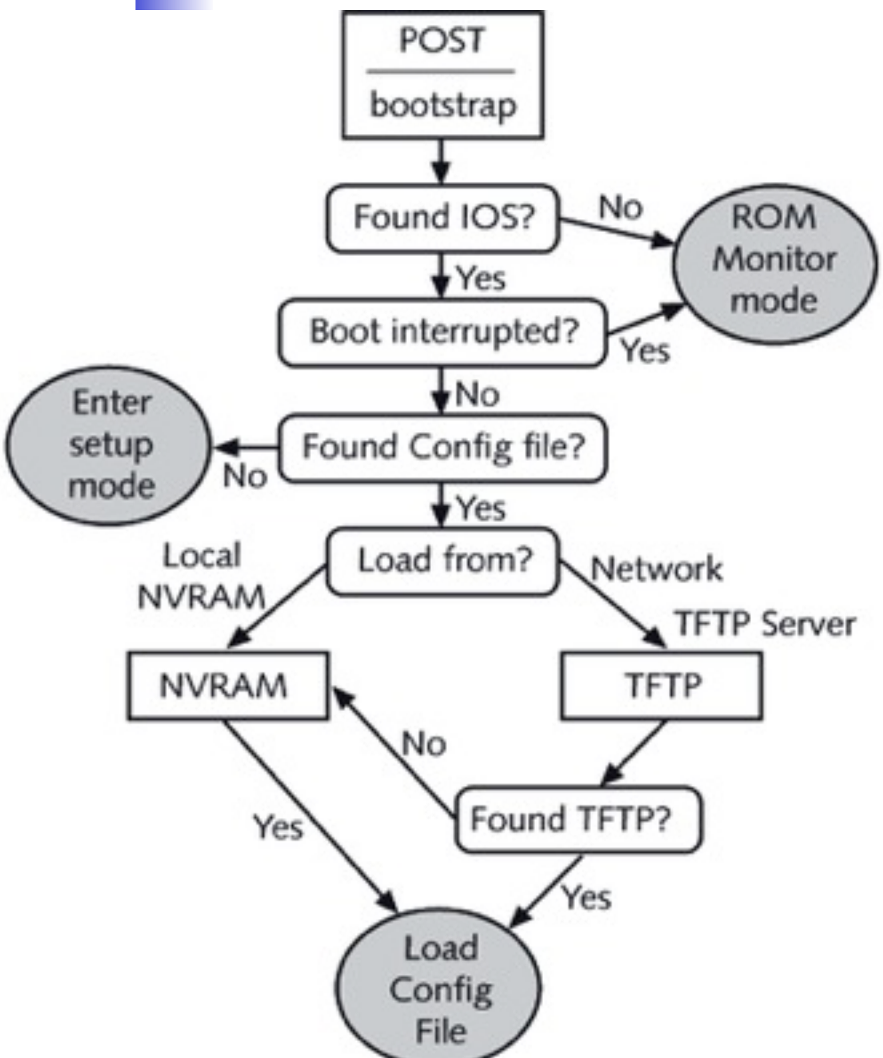
Router Modes Changed With Config-Register

- Reasons why you would want to modify the config-register:
  - Force the router into ROM Monitor Mode
  - Select a boot source and default boot filename
  - Enable/Disable the Break function
  - Control broadcast addresses
  - Set console terminal baud rate
  - Load operating software from ROM
  - Enable booting from a TFTP server

Router Modes Change and Prompts



Router Setup and Startup



- POST – loaded from ROM and runs diagnostics on hardware
- Bootstrap – locates and loads the IOS image; default is flash
- IOS – locates and loads a valid configuration from NVRAM; from startup-config and only exists if you copy running-config to NVRAM
- Startup-config – if found, router loads it and runs embedded configuration; if not found, router enters setup mode

Where is the Cisco IOS Configuration?

Command from Enable Mode	Description
copy running-config tftp	Copies the running configuration located in RAM to a TFTP server.
copy startup-config tftp	Copies the startup configuration located in NVRAM to a TFTP server.
copy tftp running-config	Copies the configuration from the TFTP server to the running configuration. The reconfiguration of the router is immediate when this command is issued. The running-config is not replaced. The files are blended.
copy tftp startup-config	Copies the configuration from the TFTP server to the startup configuration. The startup-config is replaced with the one from the TFTP server.
copy run start	Copies the working configuration file in RAM to the startup configuration file in NVRAM. Replaces the startup configuration file.
copy start run	Copies the startup configuration file in NVRAM to the running configuration in RAM. Does not replace the file in RAM; the files are blended.
copy flash tftp	Copies the IOS in flash memory to a TFTP server.
copy tftp flash	Copies the IOS from a TFTP server to flash memory.
configure terminal	Used to specify that you would like to configure your settings manually from the console terminal.
configure memory	Used to specify that you would like to pull your configuration information from NVRAM.

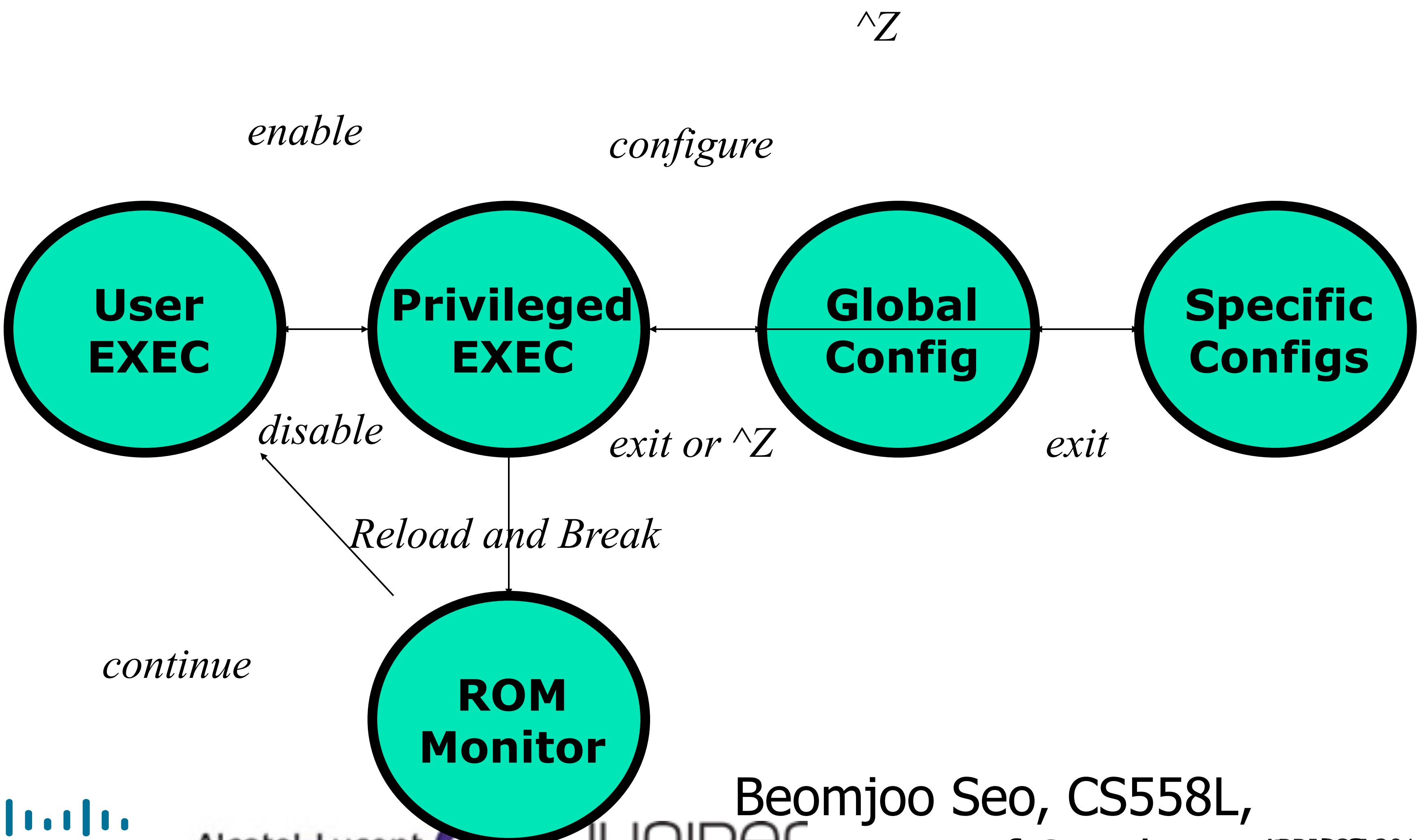
Router Access Modes

- User EXEC mode - limited examination of router
  - Router>
- Privileged EXEC mode - detailed examination of router, debugging, testing, file manipulation
  - Router#
- ROM Monitor - useful for password recovery & new IOS upload session
- Setup Mode – available when router has no `startup-config` file

Router Access Modes

Mode	Prompt	To enter	To exit	Used for
User EXEC	Router>	If there is a line password, enter it. Otherwise, press the Return or Enter key.	Logout or Exit	Shows the status of the router and allows network operators to manage connections
Privileged EXEC	Router#	Type enable at the prompt.	Disable Exit Logout	Copies, erases, sets up, and shows router settings
Global configuration	Router (config)#	Configure	Exit End	Allows you to configure various items, including clock, hostname, enable password, and enable secret password
Interface configuration	Router (config-if)#	Interface Ethernet0 *or* Interface Serial0	Exit End	Allows you to configure the settings, such as IP, for a specific interface
Line configuration	Router (config-line)#	Line console 0 *or* Line vty 0 4 *or* Line aux 0	Exit End	Configures lines, such as the console, virtual terminal, or auxiliary
Router configuration	Router (config-router)#	Router rip *or* Router igrp	Exit End	Adds or configures RIP, IGRP, or other routing protocols

CLI Modes for Router Access



External Configuration Sources

- Console – direct PC serial access
- Auxilliary port – Modem access
- Virtual terminals – Telnet access
- TFTP Server – copy configuration file into router RAM
- Network Management Software - CiscoWorks

Telnet

- Utility that connects at the highest layer of the OSI model
- Provides remote access to other devices
- Cisco routers allow telnet connections via their virtual terminal ports
- If you can establish telnet connectivity to a router, you have established that it is available on the network and that you have connectivity at all layers

SSH

- Replaces telnet for a protected command and control communication channel
- Privacy and integrity provided through the use of strong cryptographic algorithms
- Supports TACACS+, RADIUS and local authentication
- Secure Copy (SCP) available in new SSH enabled code
- Restrict access to ssh via "transport input ssh" command
- SSHv2 now in Cisco IOS!

IP Host Names

- When telnetting to a remote router or host, the IP address of the host must follow the telnet command
- Rather than using IP addresses, it is easier to refer to a remote host or router using a name
- Sometimes, you cannot gain connectivity because the host name that you are trying to connect with is entered in a table incorrectly
- Using a name server provides name resolution from one location, making a table configuration on each device unnecessary

Ping and Trace

- If you can't get connectivity at the Application layer, try connectivity at the Internetwork layer
- Ping and trace verify connectivity at the Internetwork layer
  - Both use ICMP messages to verify the destination host is reachable, and if not, give possible reasons for the problem
- Ping sends a packet to the destination and waits for a response
  - By default, the ping utility with Cisco routers is configured to send five packets to the target

Ping and Trace

- Extended mode ping
  - Options include:
    - The destination address of the ping
    - The protocol
    - Repeat count
    - Datagram size
  - Can only be accessed from the privileged mode prompt

Ping and Trace

- If ping indicates a problem with connectivity, using trace may provide a better clue as to the source of the connectivity problem
- Trace command is similar to ping command, except that the replies are requested at each hop along the way to the destination
- Trace sends multiple ICMP packets with progressively higher TTL counters until the packet reaches the destination

IP Route

- If you cannot get connectivity using ping or trace, you should check your routing table
- You can issue the show ip route command from the enable mode prompt
  - This command shows the routing table
- Typically, routing tables are dynamically created when routing protocols are configured on the router

Checking the Interface

- On of the biggest mistakes made when troubleshooting is not checking the interfaces on the router
- If the interfaces are down, packets cannot be delivered
- Router interfaces go down for a variety of reasons including:
  - Incorrect IP configuration
  - Cable problems

Checking the Interface

- Keepalive frames
  - Data frames sent between two hosts to ensure that the connection between those hosts remains open
- Different types of interfaces can show different types of reports
  - For example, a Token Ring interface reports down when there is no electrical carrier signal present

Checking the Interface

```
lab-a#show interfaces
Ethernet0 is up, line protocol is up
  Hardware is Lance, address is 0000.0c8e.b490 (bia 0000.0c8e.b490)
  Internet address is 192.5.5.1/24
  MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input never, output 00:00:03, output hang never
  Last clearing of "show interface" counters never
  Queueing strategy: fifo
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
Serial0 is down, line protocol is down
  Hardware is HD64570
  Internet address is 201.100.11.1/24
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, loopback not set
  Keepalive set (10 sec)
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Queueing strategy: fifo
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     15 packets output, 3198 bytes, 0 underruns
     0 output errors, 0 collisions, 2 interface resets
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions
     DCD=up  DSR=up  DTR=down  RTS=down  CTS=up
Serial1 is administratively down, line protocol is down
  Hardware is HD64570
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, loopback not set
  Keepalive set (10 sec)
  Last input never, output never, output hang never
```

Interface E0 is fully functional. Frames can be sent and received on this interface.

S0 is not functional. In this case, the serial interface on the router attached to this router is down. If one end of a point-to-point link is down, it will "push" the attached up interface on the next router down.

The S1 interface is not functional. In this case, there is no cable attached to S1 as it is not being used.

Clear Counters

- Routers keep detailed statistics regarding the data passing across its interfaces
- Before using the show interface command, you may want to clear the existing interface information
- You can clear these statistics (**counters**) on the interface by using the clear interface or clear counters command

Debug

- Debug command
  - One of the most powerful tools you can use to obtain information from your router
  - Only available from privileged EXEC mode
  - Has numerous subcommands that allow you to troubleshoot particular protocols
  - Allows you to check for specific types of traffic on the wire

Debug

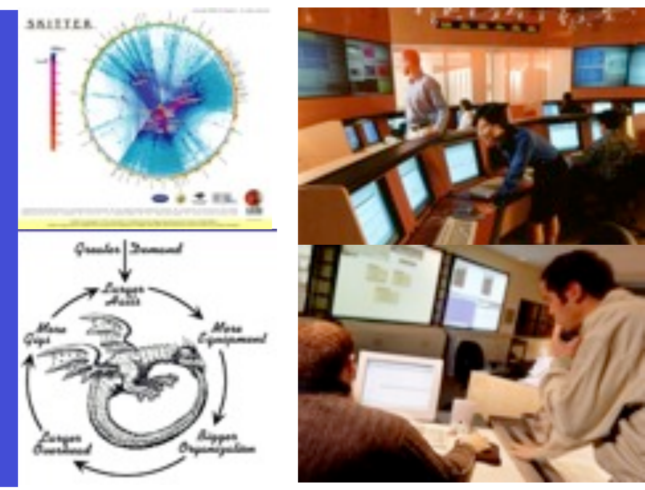
```
RouterB#debug all
This may severely impact network performance. Continue? [confirm]

All possible debugging has been turned on
RouterB#
IP: s=172.22.3.1 (Serial1), d=255.255.255.255, len 76, rcvd 2
UDP: rcvd src=172.22.3.1(520), dst=255.255.255.255(520), length=52
RIP: received v1 update from 172.22.3.1 on Serial1
     172.22.4.0 in 1 hops
     172.22.5.0 in 2 hops
RIP: Update contains 2 routes
SERVICE_MODULE(0): lxt441 interrupt 1 status A7 loop 0
SERVICE_MODULE(0): lxt441 interrupt 1 status 87 loop 0
SERVICE_MODULE(0): lxt441 interrupt 1 status A7 loop 0
SERVICE_MODULE(0): lxt441 interrupt 1 status 87 loop 0
Serial1: HDLC myseq 6631, mineseen 6631, yourseen 6580, line up
SERVICE_MODULE(0): lxt441 interrupt 1 status A7 loop 0
SERVICE_MODULE(0): lxt441 interrupt 1 status 87 loop 0
SERVICE_MODULE(0): lxt441 interrupt 1 status A7 loop 0
SERVICE_MODULE(0): lxt441 interrupt 1 status 87 loop 0
RIP: sending v1 update to 255.255.255.255 via Ethernet0 (172.22.2.1)
     subnet  172.22.3.0, metric 1
     subnet  172.22.4.0, metric 2
     subnet  172.22.5.0, metric 3
RIP: Update contains 3 routes
IP: s=172.22.2.1 (local), d=255.255.255.255 (Ethernet0), len 55, sending broad/m
ulticast
RIP: sending v1 update to 255.255.255.255 via Serial1 (172.22.3.2)
     subnet  172.22.2.0, metric 1
RIP: Update contains 1 routes
IP: s=172.22.3.2 (local), d=255.255.255.255 (Serial1), len 67, sending broad/mul
ticast
SERVICE_MODULE(0): lxt441 interrupt 1 status A7 loop 0
SERVICE_MODULE(0): lxt441 interrupt 1 status 87 loop 0
Serial0: attempting to restart
Serial1: HDLC myseq 6632, mineseen 6632, yourseen 6581, line up
IP: s=172.22.5.1 (Ethernet0), d=255.255.255.255, len 106, rcvd 2
UDP: rcvd src=172.22.5.1(520), dst=255.255.255.255(520), length=72
RIP: ignored v1 update from bad source 172.22.5.1 on Ethernet0
SERVICE_MODULE(0): lxt441 interrupt 1 status A7 loop 0
SERVICE_MODULE(0): lxt441 interrupt 1 status 87 loop 0
SERVICE_MODULE(0): lxt441 interrupt 1 status A7 loop 0
SERVICE_MODULE(0): lxt441 interrupt 1 status 87 loop 0
SERVICE_MODULE(0): lxt441 interrupt 1 status A7 loop 0
SERVICE_MODULE(0): lxt441 interrupt 1 status 87 loop 0
Serial1: HDLC myseq 6633, mineseen 6633, yourseen 6582, line up
All possible debugging has been turned off
RouterB#
```

The debug all command warns you that issuing this command could cause severe network congestion. This command should only be used for a short period of time as a troubleshooting tool.

LAB

- Follow the lab-guide to set up initial topology

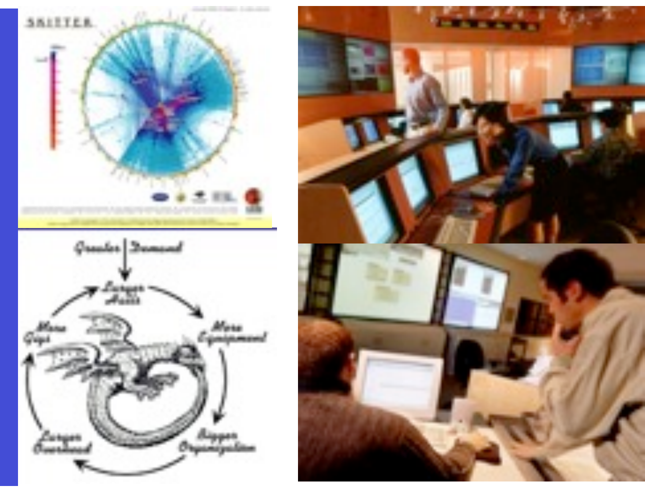
Infrastructure Security



Infrastructure Security

- Best Common Practice [BCP]
  - Secure Router Access
  - Edge Protection
  - Remote triggered black hole filtering
  - Sink holes
  - Source address validation on all customer traffic
  - Control Plane Protection

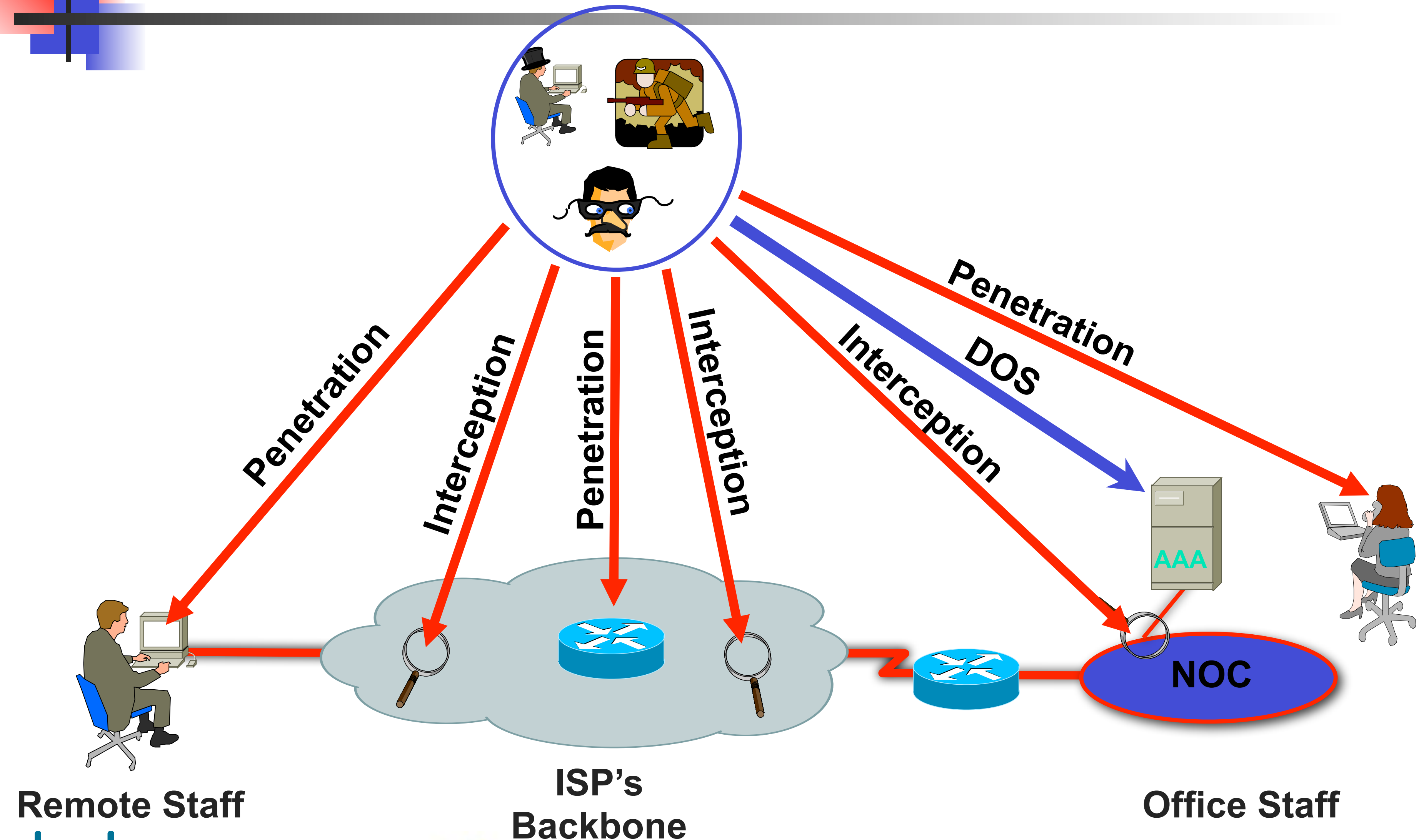
Secure Router Access



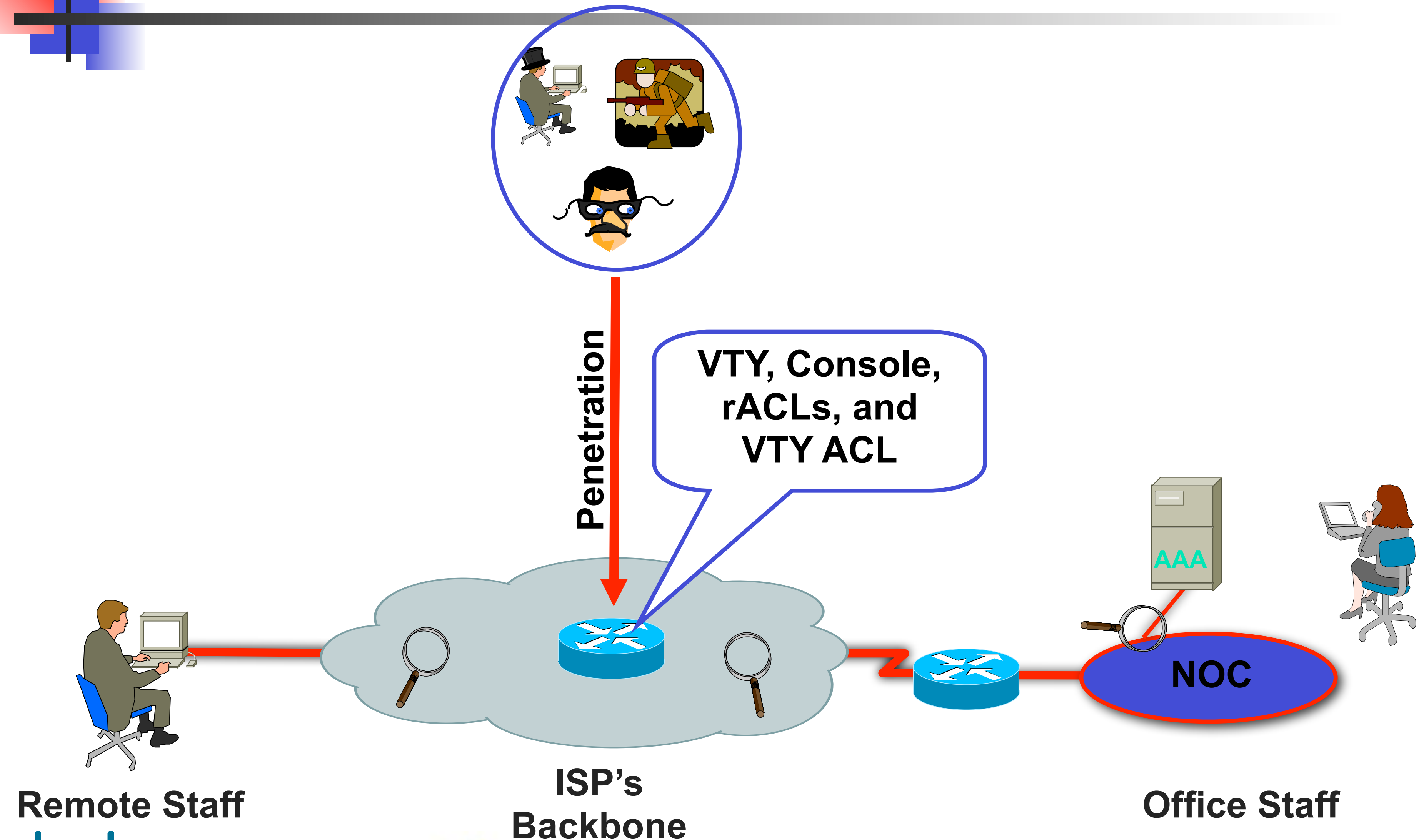
Check List

- AAA to the Network Devices
- Controlling Packets Destined to the Network Devices
- Config Audits

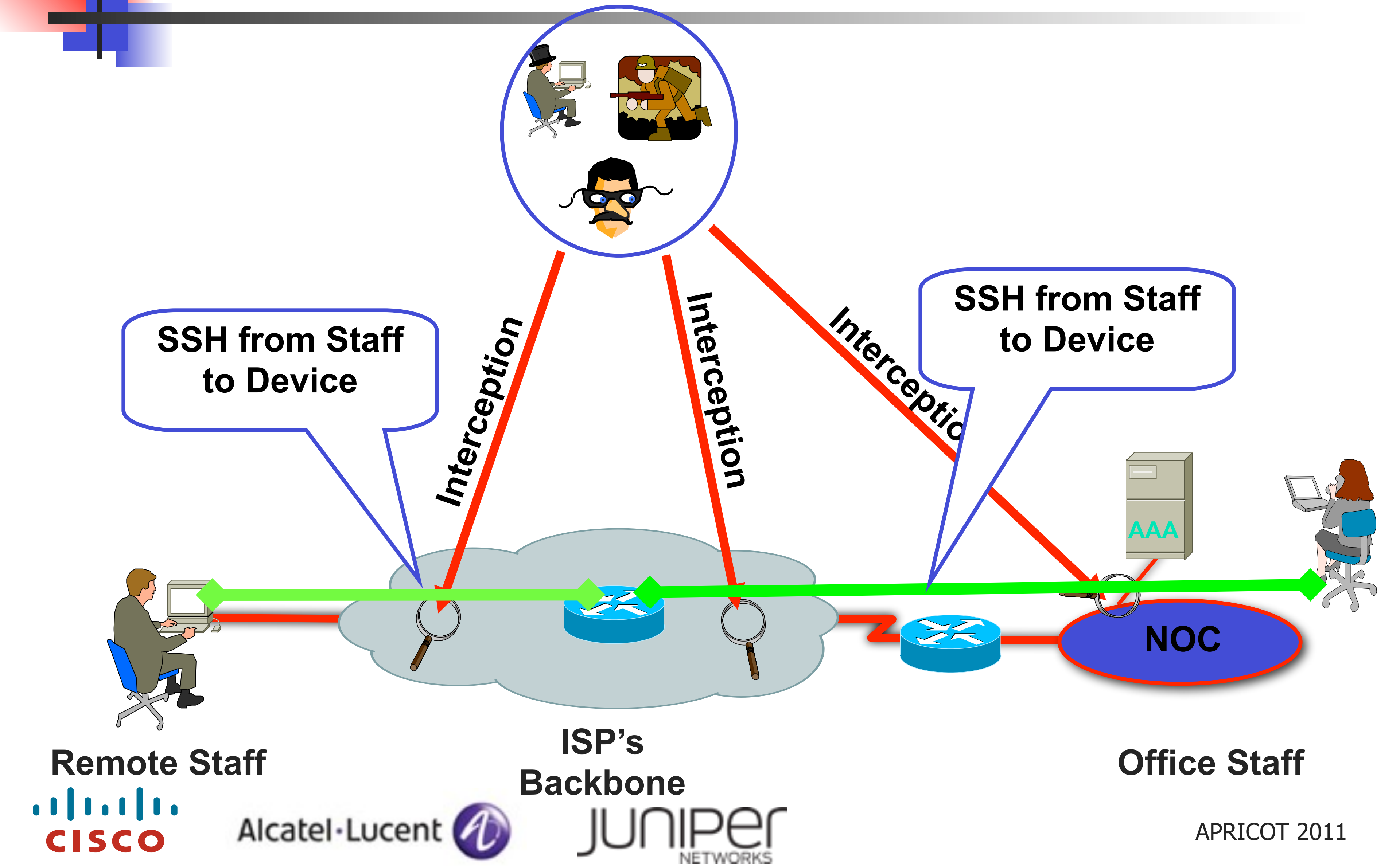
RISK Assessment



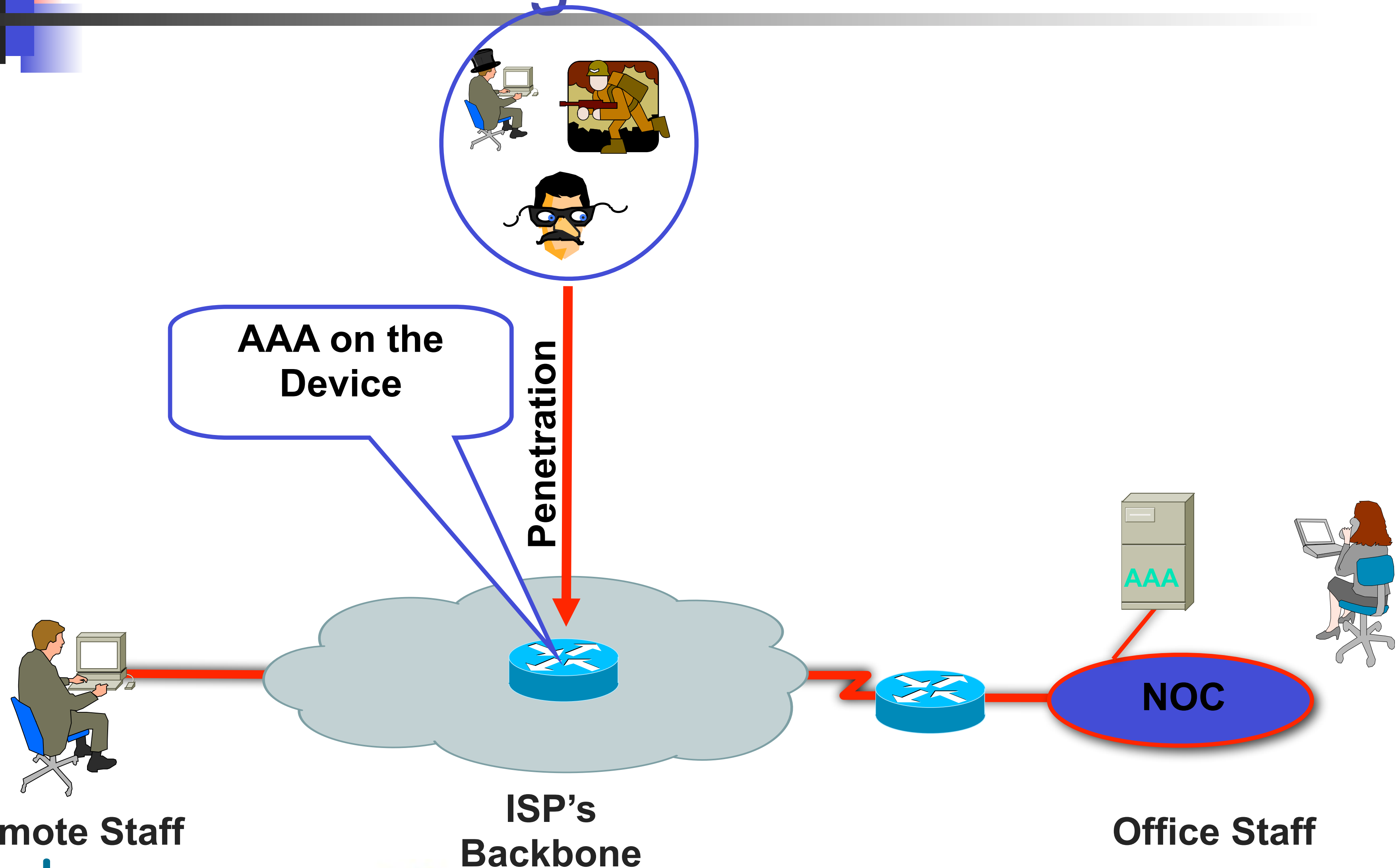
Lock Down the VTY and Console Ports



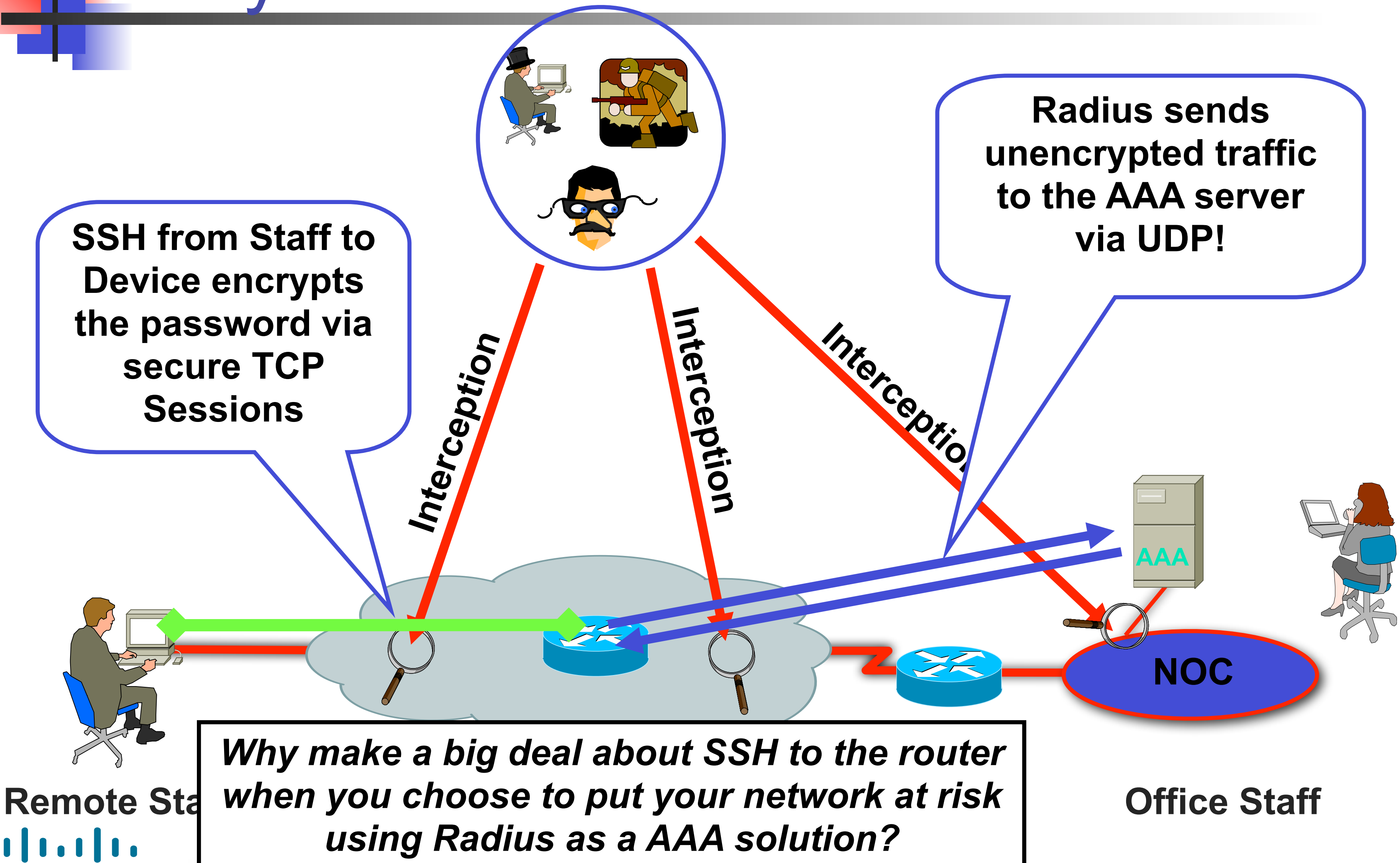
Encrypt the Traffic from Staff to Device



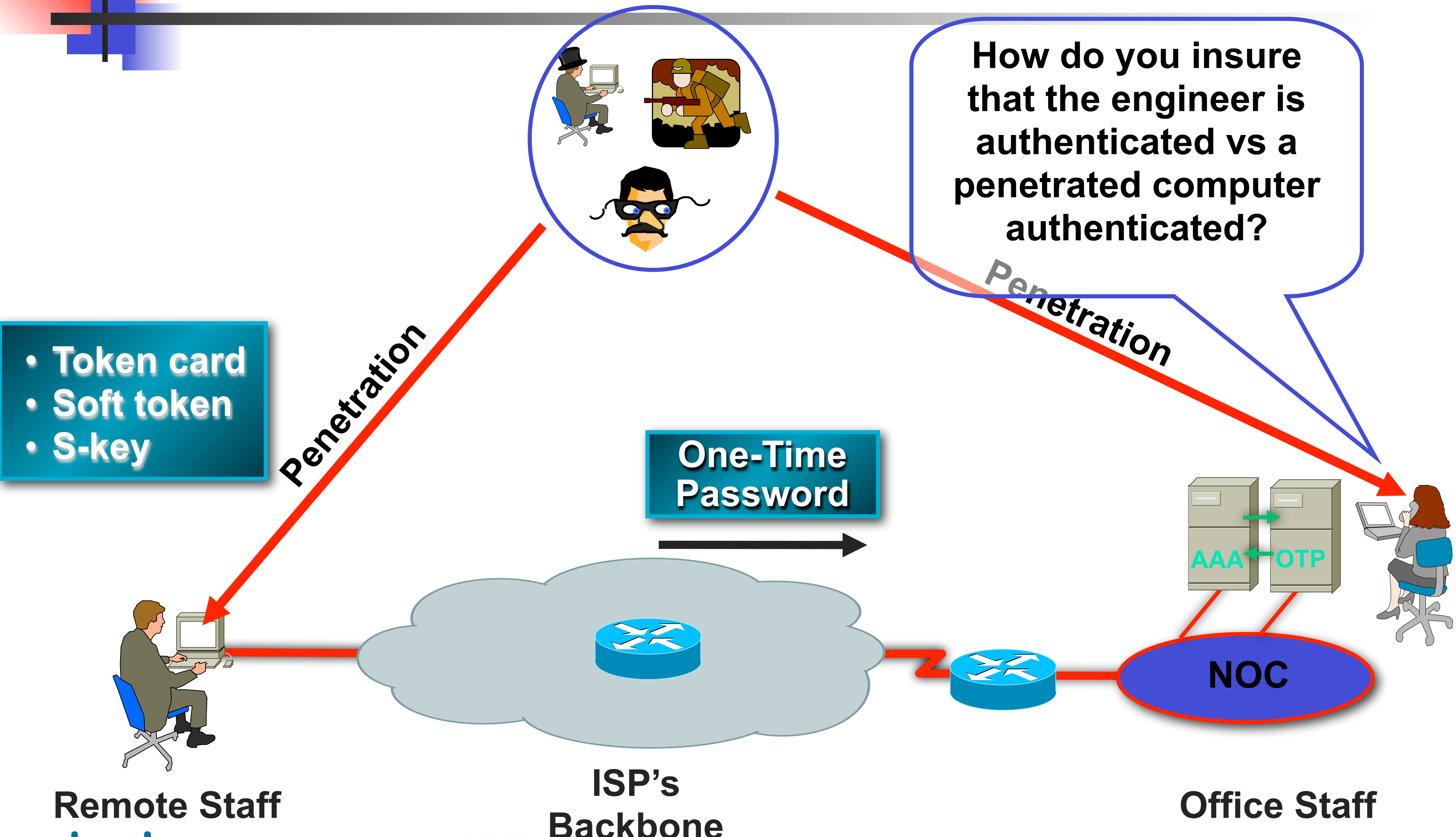
Staff AAA to get into the Device



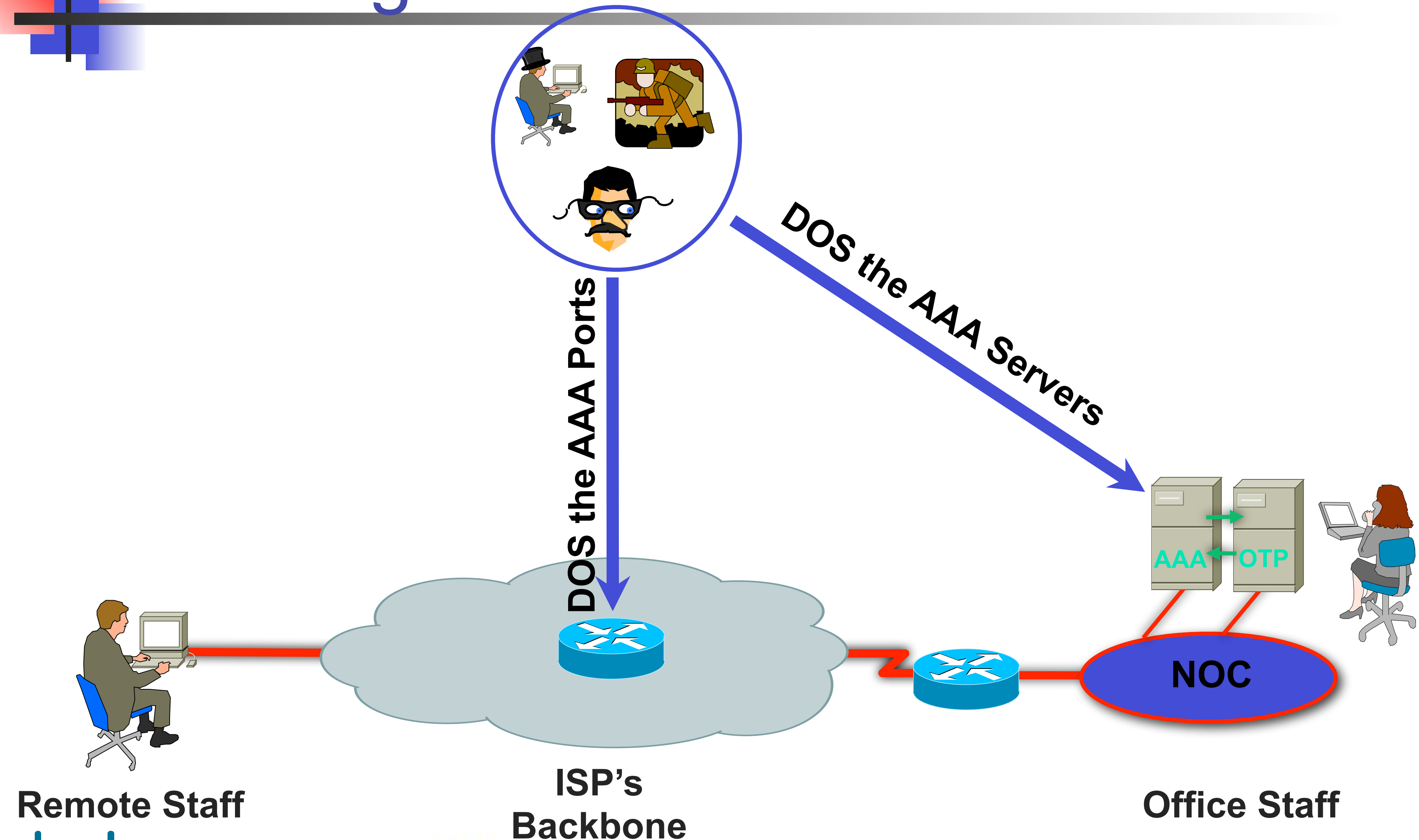
Radius is not an ISP AAA Option! Why?



One Time Password – ID Check

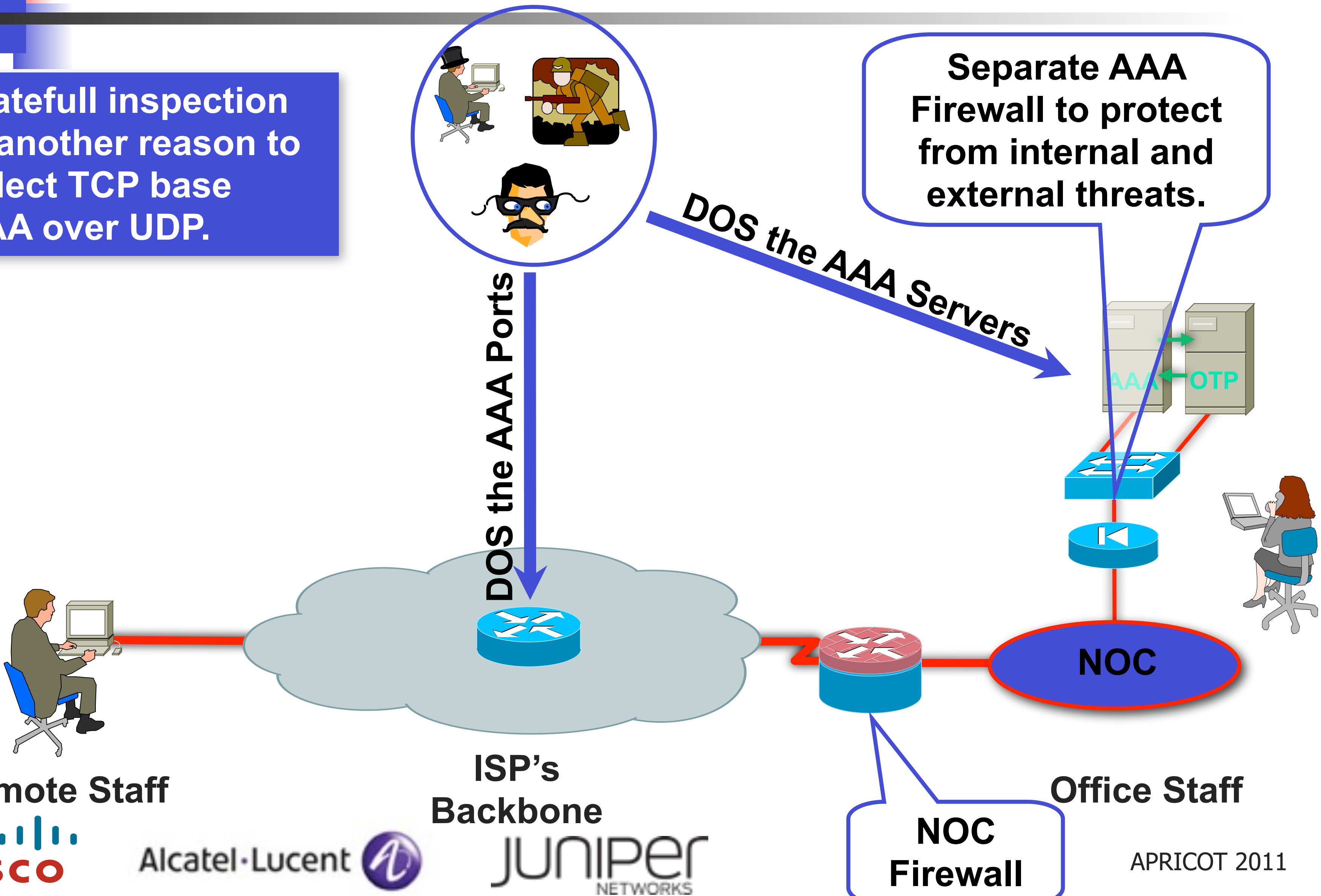


DOSing the AAA Infrastructure

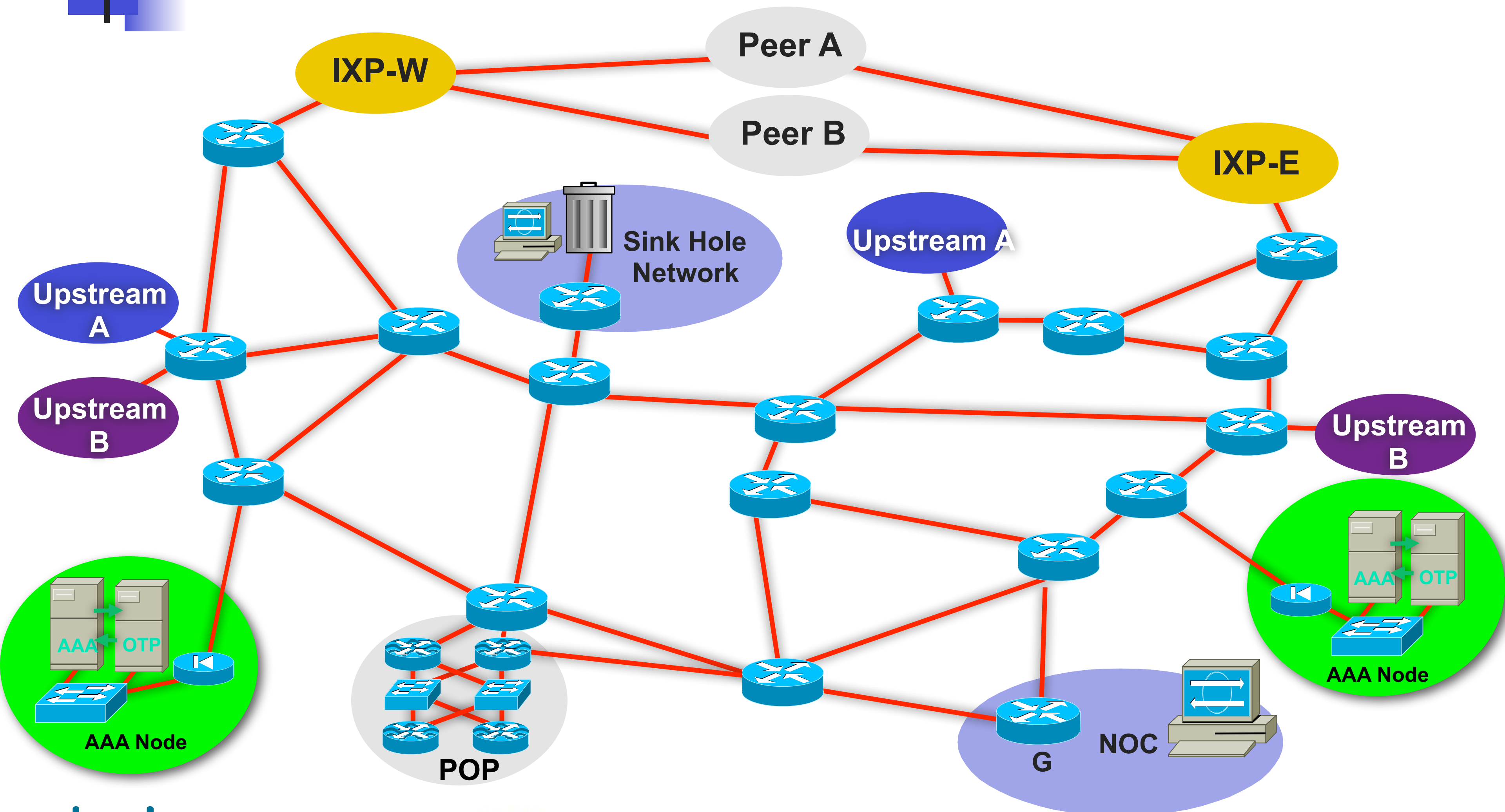


Use a Firewall to Isolate the AAA Servers

Statefull inspection is another reason to select TCP base AAA over UDP.



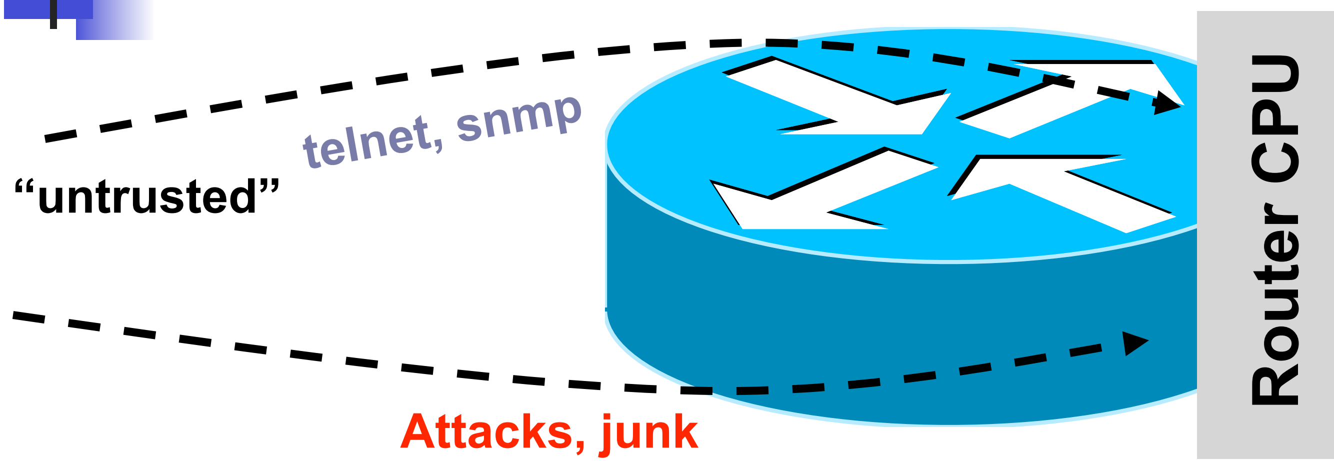
Distribute AAA Servers and Config Backup



TACACS+ URLs

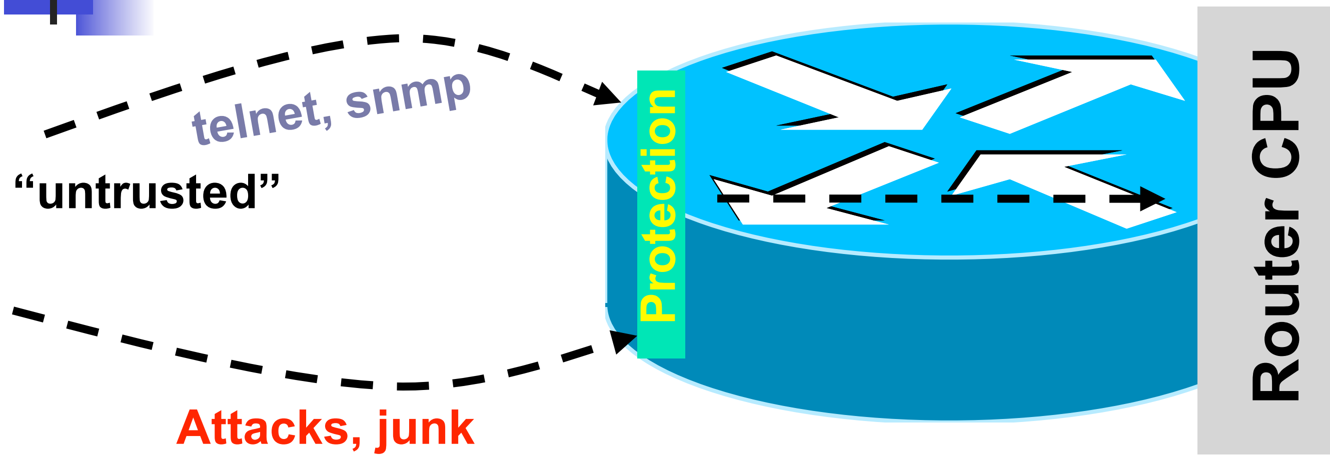
- TACACS+ Open Source
  - ftp://ftp-eng.cisco.com/pub/tacacs/
  - Includes the IETF Draft, Source, and Specs.
- Extended TACACS++ server
  - http://freshmeat.net/projects/tacpp/
- TACACS + mods
  - http://www.shrubbery.net/tac_plus/

The Old World: Router Perspective



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

The New World: Router Perspective



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

Watch the Config!

- There has been many times where the only way you know someone has violated the router is that a config has changed.
- Of course you need to be monitoring your configs.

Config Monitoring

- **RANCID - Really Awesome New Cisco config Differ (but works with lots of routers)**
  - **http://www.shrubbery.net/rancid/**
  - **http://www.nanog.org/mtg-0310/rancid.html**
- **Rancid monitors a device's configuration (software & hardware) using CVS.**
- **Rancid logs into each of the devices in the device table file, runs various show commands, processes the output, and emails any differences from the previous collection to staff.**

Controlling access

Securing the box

- Identify what services to what networks
  - Classic prudent policy
  - Allow what you know
  - Then deny all!

Securing Physical Access

- No method of configuring a system can protect it if physical access is not secure
- Attackers with physical access can:
  - Physically harm and degrade the system
  - Perform password recovery and obtain access to the CLI
  - Attach a tap and packet sniffer to obtain traffic captures
  - Do other nefarious things limited only by their imagination…

Protect your systems from unauthorized physical access!

Protecting the Diagnostics Port

```
[[edit system]
lab@r5# set diag-port-authentication ?
Possible completions:
+ apply-groups           Groups from which to inherit configuration data
  encrypted-password     Crypted password string
  plain-text-password    Prompt for plain text password (auto-crypted)
[edit system]
```

- Passwords
  - Use `plain-text-password` to enter password directly
    - Stored as an MD5 hash in the configuration
  - Use `encrypted-password` to paste in an existing MD5 hash

Securing Logical Access

- Logical access is more difficult to secure
  - Attackers do not require physical access to logically access the system
- Terminal servers can provide backdoor. For you or an attacker
  - Some extra logins/ features are available from console
- One of many ways to protect your systems from unauthorized logical access is to secure it with user names and passwords

Agenda: Configuring Root Authentication

- Why Secure CLI Access Is Needed
- → Configuring Root Authentication
- Configuring Login Users and Classes
  - Creating Users
  - Creating Login Classes
  - Setting the Idle Timeout

Root Authentication

- By default, Juniper Networks routers have only a single user configured, called root
  - Juniper Networks routers do not have a default password configured for the root account.
- Cisco has no console login password or enable password
- Systems with root accounts with no passwords do not last long on the Internet

Configure the root account with a difficult-to-guess password as the first step in securing the system!

Agenda: Configuring Login Users and Classes

- Why Secure CLI Access Is Needed
- Configuring Root Authentication
- → Configuring Login Users and Classes
  - Creating Users
  - Creating Login Classes
  - Setting the Idle Timeout

Creating Users

```
[edit]
lab@R5# edit system login user jsmith

[edit system login user jsmith]
lab@R5#
```

Adding a user involves only creating the appropriate container in the configuration

Setting the Full Name

```
[edit system login user jsmith]
lab@R5# set full-name "John Smith"

[edit system login user jsmith]
lab@R5# show
full-name "John Smith";
```

Optionally, you can configure a text string to identify this user

Setting the Password

```
[edit system login user jsmith]
lab@R5# set authentication plain-text-password
New password: extremely tough password
Retype new password: extremely tough password

[edit system login user jsmith]
lab@R5# show
full-name "John Smith";
authentication {
    encrypted-password
  "$1$wrcIE7//$61gsASq1vP9OktkPgpiCz0"; # SECRET-
  DATA
}
```

Stored in the configuration as an MD5 hashed value

Attaching Users to Login Classes

```
[edit system login user jsmith]
lab@R5# set class ?
Possible completions:
  <class>                    Login class
  operator
  read-only
  super-user
  superuser
  unauthorized
```

- A user must be a member of one (and only one) login class
  - Preconfigured login classes are available

Preconfigured Login Classes

Class	Permission Bits Set
super-user superuser (Identical functionality)	All
read-only	View
operator	Clear, Network, Reset, Trace, View
unauthorized	None

Available Permissions (1 of 2)

```
[edit system login class restricted-operator]
lab@R5# set permissions ?
Possible completions:
  [                      Open a set of values
  admin                  Can view user accounts
  admin-control          Can modify user accounts
  all                    All permission bits turned on
  clear                  Can clear learned network information
  configure              Can enter configuration mode
  control                Can modify any configuration values
  edit                   Can edit full files
  field                  Special for field (debug) support
  firewall               Can view firewall configuration
  firewall-control       Can modify firewall configuration
  floppy                 Can read and write the floppy drive
  interface              Can view interface configuration
  interface-control      Can modify interface configuration
```

Available Permissions (2 of 2)

maintenance	Can perform system maintenance (as wheel)
network	Can access the network
reset	Can reset and restart interfaces and
rollback	Can rollback for depth greater than zero
routing	Can view routing configuration
routing-control	Can modify routing configuration
secret	Can view secret configuration
secret-control	Can modify secret configuration
security	Can view security configuration
security-control	Can modify security configuration
shell	Can start a local shell
snmp	Can view SNMP configuration
snmp-control	Can modify SNMP configuration
system	Can view system configuration
system-control	Can modify system configuration
trace	Can view trace file settings
trace-control	Can modify trace file settings

Allow and Deny Commands

```
[edit system login class restricted-operator]
lab@R5# set allow-commands "clear bgp"

[edit system login class restricted-operator]
lab@R5# set deny-commands "telnet"

[edit system login class restricted-operator]
lab@R5# show
permissions [ network trace view ];
allow-commands "clear bgp";
deny-commands telnet;
```

- More options:
  - Allow commands permit additional access beyond that allowed by permissions
  - Deny commands restrict access normally allowed by permissions

Regular Expressions—Commands

```
[edit system login class restricted-operator]
lab@R5# set allow-commands "clear ospf|clear bgp"

[edit system login class restricted-operator]
lab@R5# show
permissions [ network trace view ];
allow-commands "clear ospf|clear bgp";
deny-commands telnet;
```

Operator	Match …
|	One of the two terms
^	Beginning of the expression
$	End of the expression
[ ]	Range of letters or digits
( )	Group of commands

Regular Expressions—Example 1

```
[edit]
 system {
     login {
          class operator-may-reboot {
               permissions [ clear network reset trace view ];
               allow-commands "request system reboot";
          }
}
```

Class has operator privileges and can reboot the system

Operator	Match …
|	One of the two terms
^	Beginning of the expression
$	End of the expression
[ ]	Range of letters or digits
( )	Group of commands

Regular Expressions—Example 2

```
[edit]
 system {
    login {
        class may-not-set {
              permissions [ clear network reset trace view ];
              deny-commands "^set";
         }
}
```

Class has operator privileges but cannot use any command beginning

Operator	Match …
|	One of the two terms
^	Beginning of the expression
$	End of the expression
[ ]	Range of letters or digits
( )	Group of commands

Regular Expressions—Configuration

```
[edit system login class restricted-configuration]
lab@R5# set deny-configuration "(system login class) | (system services)"
[edit system login class restricted-configuration]
lab@R5# show
permissions configure;
allow-commands "(show bgp summary | show ospf neighbor)";
```

Operator	Match ...
|	One of the two terms
^	Beginning of the expression
$	End of the expression
[ ]	Range of letters or digits
( )	Group of commands

IOS Role-Based CLI Views commands

- Enable AAA using the 'aaa new-model' global config command
  - aaa new-model
- Configure the AAA default list to use the router's local database for authentication and authorization
  - aaa authentication login default local
  - aaa authorization exec default local
- Configure AAA console authorization
  - aaa authorization console
- Access the root view. You need to first configure a secret or enable password before you can access the root view. configure a secret password = cisco)
  - Edge_C38#conf t
- Enter configuration commands, one per line. End with CNTL/Z.
  - Edge_C38(config)#enable secret cisco
  - Edge_C38(config)#^Z
  - Edge_C38#
  - Edge_C38#enable view
  - Password: secret_password

IOS Role-Based CLI Views commands

- Edge_C38#
- Create the Operator view
- Configure a password for this view
  - Ping
  - Show controllers
  - Show interfaces
  - Show version
- parser view operator
  - password 5 opspassword
  - commands exec include ping
  - commands exec include show version
  - commands exec include show controllers
  - commands exec include show interfaces

Setting the Idle Timeout

```
[edit system login class restricted-operator]
lab@R5# set idle-timeout 10
```

JUNOS

```
R5(config)#  line  vty o 4

R5(config)#  exec timeout 0 10
```

IOS

- No idle-timeout by default
  - Set the time, in minutes, after which an idle user is automatically disconnected
  - User session is sent warning messages 5 minutes, 1 minute, 10 seconds, and on session shutdown

Is Any Remote Access Secure?

- No remote access to the router is completely secure
  - The issue:
    - Nothing is ever completely secure
    - We need remote access to the router
    - We must minimize the risk
  - The defaults:
    - CLI access is available only on the console port
    - No access is available on the auxiliary port without configuration
    - No other method of remote access is available without configuration

Remote Access Methods

- Remote access methods available on most platforms:
  - Remote access to the CLI
    - Telnet (Client and server)
    - SSH (Client and server)
    - Rlogin (Server, client available in the shell, JUNOS)
  - Remote access to the file system, (JUNOS)
    - FTP (client and server)
    - SCP (client and server)
  - Other remote access
    - Finger (server, client available in the shell)
    - SNMP

Telnet and FTP Servers

- Telnet and FTP
  - Provide convenient access to the CLI and file system
  - Both the Telnet and FTP Servers are disabled by default
  - Everything transmitted (including the password) is sent in cleartext on the wire
  - Custom packet sniffers are available to search for and reassemble the passwords in a both Telnet and FTP sessions
  - The root user can never log in with Telnet or FTP
  - Both protocols provide only availability
    - Confidentiality and integrity are not protected

Enabling the Telnet Server

- The following command enables Telnet access
  - Disabled by default

JUNOS

```
[edit system]
lab@R1# set services telnet

[edit system]
lab@R1# show
services {
    telnet;
```

IOS

```
R1(config)# line vty 0 4
R1(config)# login
R1(config)# password Uj%$3
```

Additional Telnet Options

Options:

- The connection limit establishes the maximum number of concurrent sessions (JUNOS default = 75) (IOS default = 5)

- The rate limit establishes the maximum number of connections allowed per minute (JUNOS default = 150)

Enabling the FTP Server

- The following command enables the FTP server
  - Disabled by default
  - **[edit system]**

```
lab@R1# set services ftp


[edit system]
lab@R1# show
services {
    ftp;
}
```

SSH Clients

- Many SSH implementations are available
  - Putty
  - TeraTerm
  - OpenSSH
  - Many commercial servers and clients

Enabling the SSH Server

- The following command enables SSH access
  - Disabled by default

```
[edit system]
lab@R1# set services ssh

[edit system]
lab@R1# show
services {
    ssh;
}
```

Allowing Root Logins

```
[edit]
lab@R1# set system services ssh root-login ?
Possible completions:
  allow                    Allow root access via ssh
  deny                     Do not allow root access via ssh
  deny-password            Allow for non-password-based authentication methods
   only
```

- By default, once SSH access is turned on, user root cannot log in with SSH
  - User root can be allowed to log in:
    - Normally, with password-based authentication
    - Only with key-based authentication
  - Is this something you really want to do?

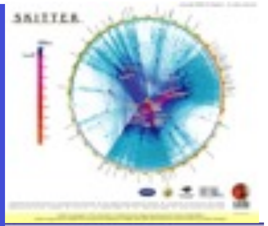
Enabling SSH - IOS

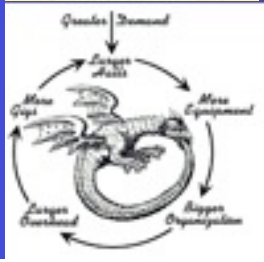
```
cry key generate rsa

ip ssh time-out 60
ip ssh authentication-retries 2

!--- Step 4: By default the vtys' transport is Telnet. In
   this case,
!--- Telnet is disabled and only SSH is supported.

line vty 0 4
transport input SSH
```

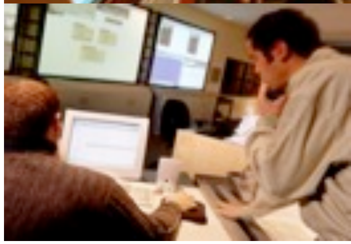
Lab – Securing remote access

- Follow the lab guide

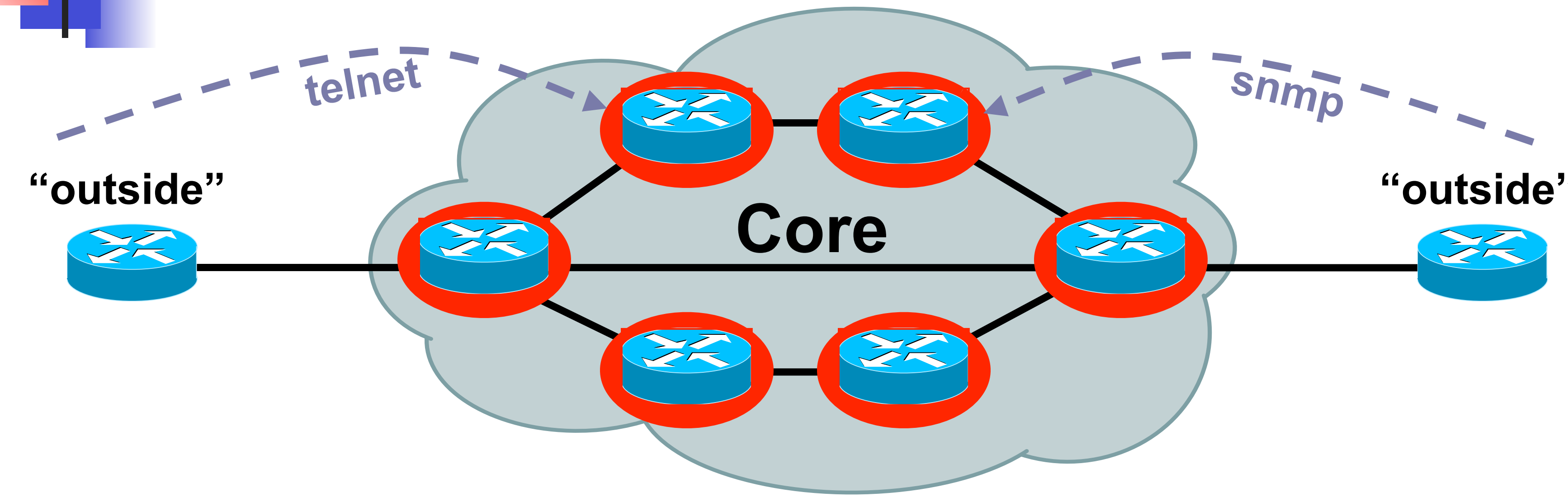
Edge Protection





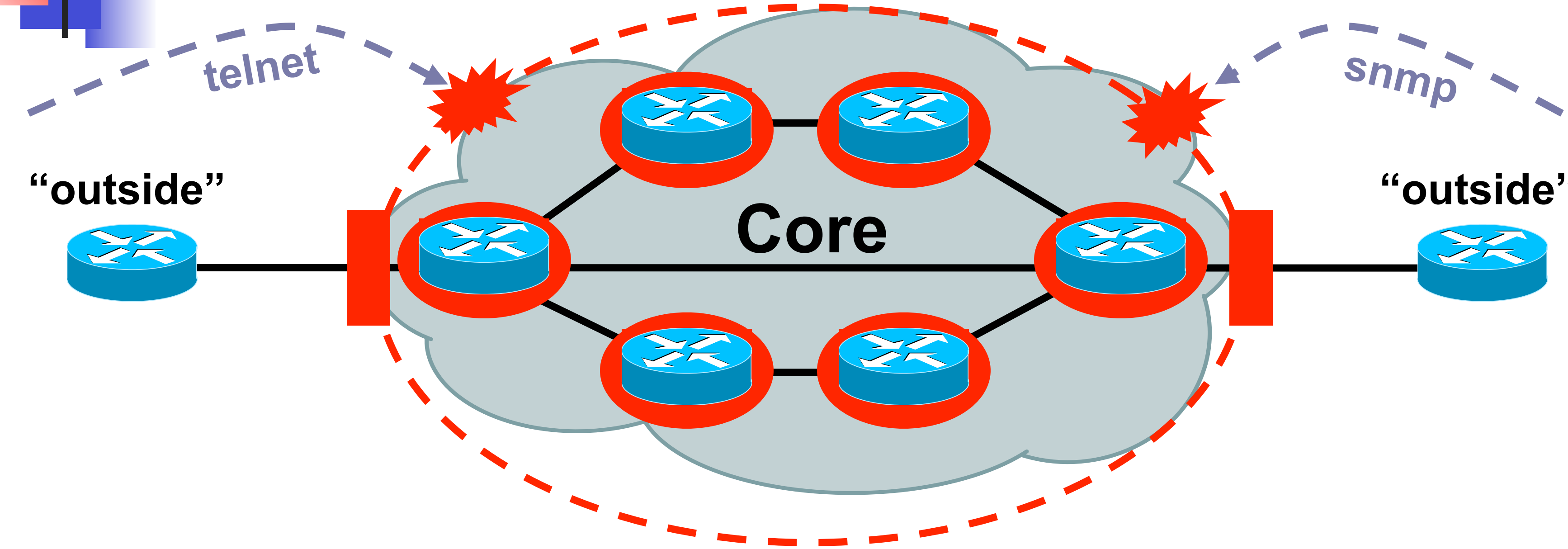


The Old World: Network Edge



- Core routers individually secured
- Every router accessible from outside

The New World: Network Edge



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

Infrastructure ACLs

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

Infrastructure ACLs

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

A Digression: IP Fragments and Security

- Fragmented Packets can cause problems...
  - Fragmented packets can be used as an attack vector to bypass ACLs
  - Fragments can increase the effectiveness of some attacks by making the recipient consume more resources (CPU and memory) due to fragmentation reassembly

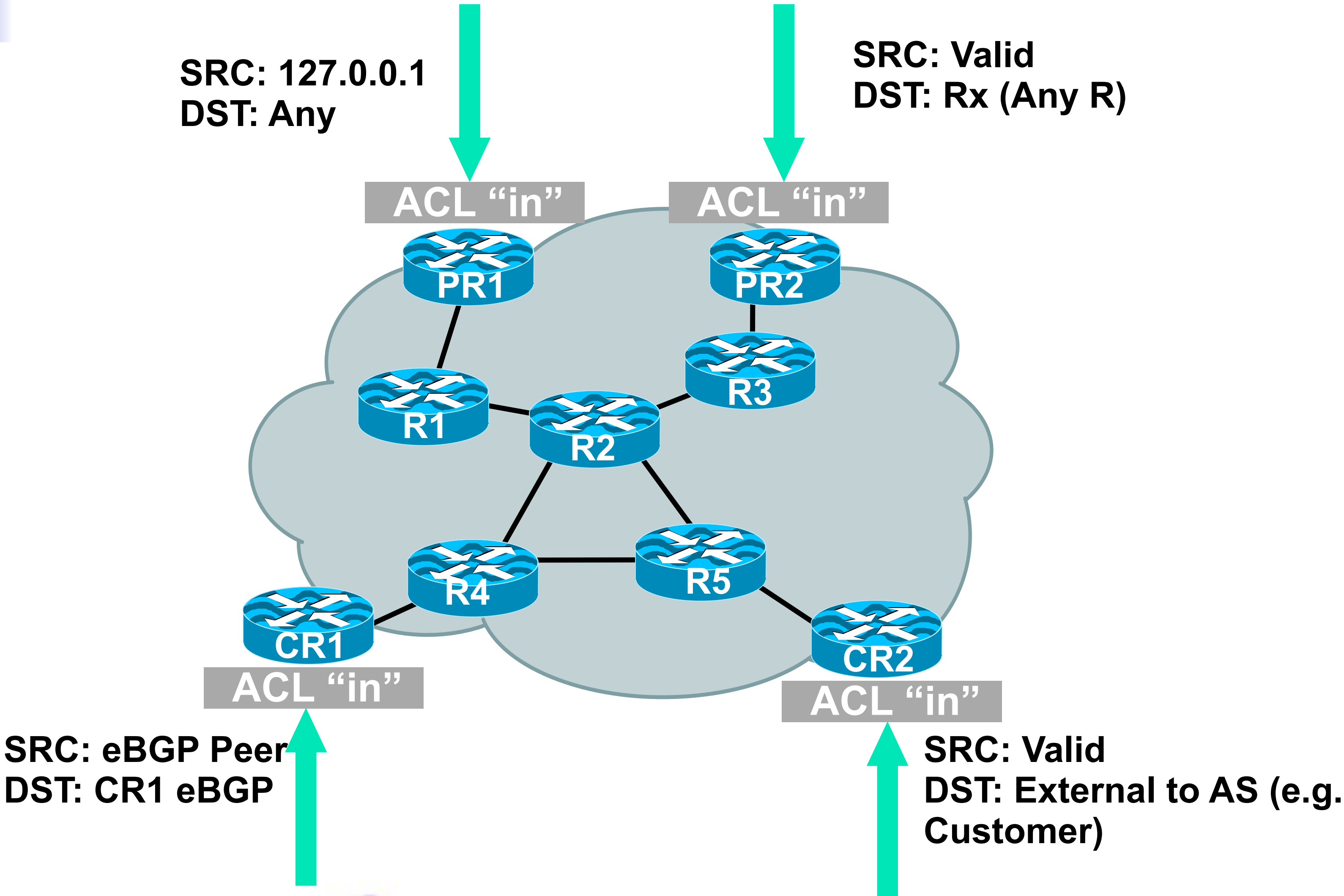
A Digression: IP Fragments and Security

- By default (without the fragments keyword)…
  - Initial fragments and non-fragmented packets
    - L3 ACLs— access control entry (ACE) action executed (permit/deny) available L3 information
    - L4 ACLs—ACE action executed (permit/deny) available L4 information
  - Non-initial fragment packets (assuming L3 match)
    - L3 ACLs—ACE action executed (permit/deny) available L3 information
    - L4 ACLs—ACE action executed (permit/deny) based on L3 info (there is no L4 info in the fragment) and protocol only
- The ACL fragments keyword enables specialized handling behavior
  - Initial fragments and non-fragmented packets
    - L3 and L4 ACLs—the packet does not match the entry since the fragment keyword is used. The packet then "falls through" to the next line(s)
  - Non-initial fragment packets (assuming L3 match)
    - With L3 and L4 ACLs—with an L3 match (and protocol matches the

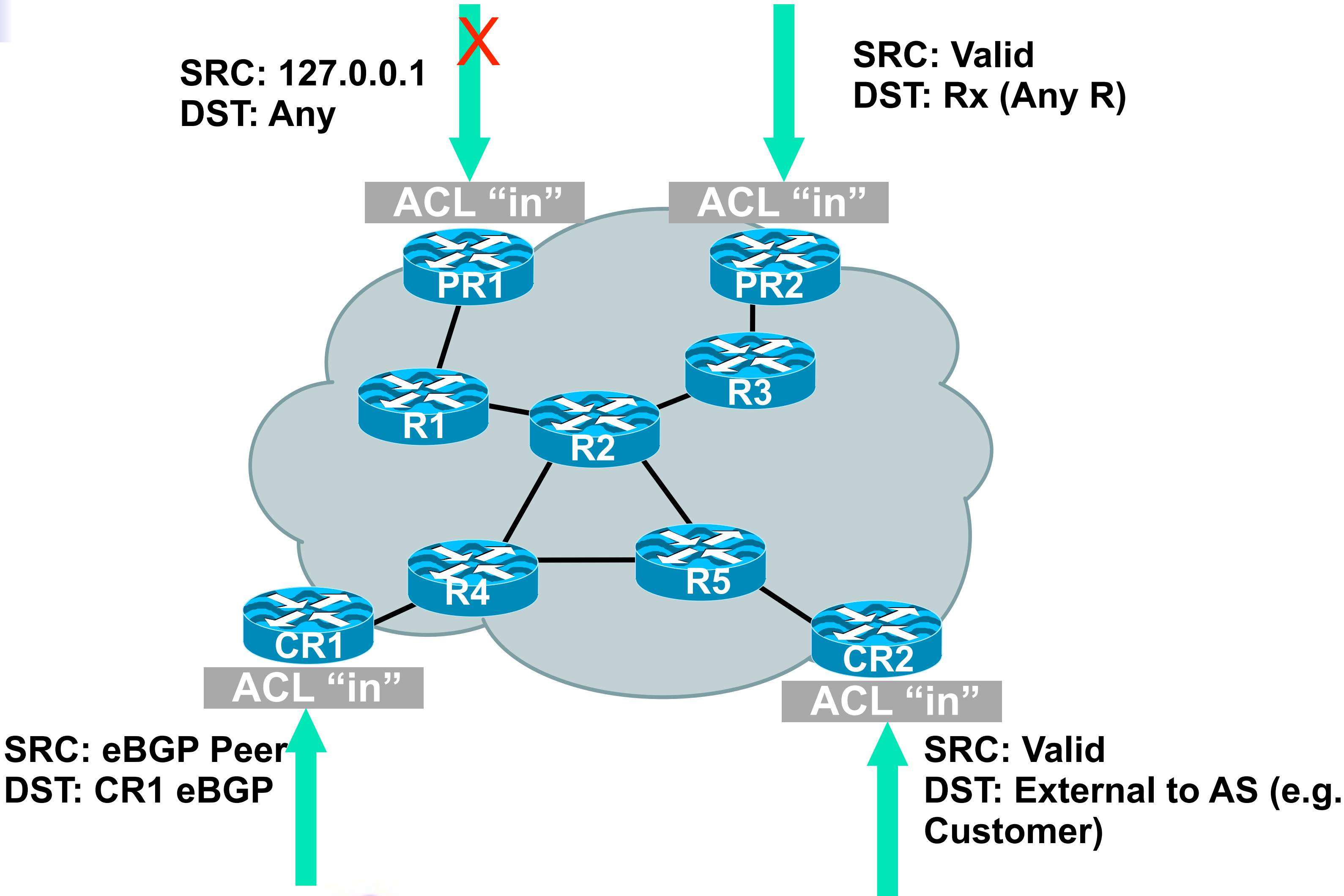
Infrastructure ACLs

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

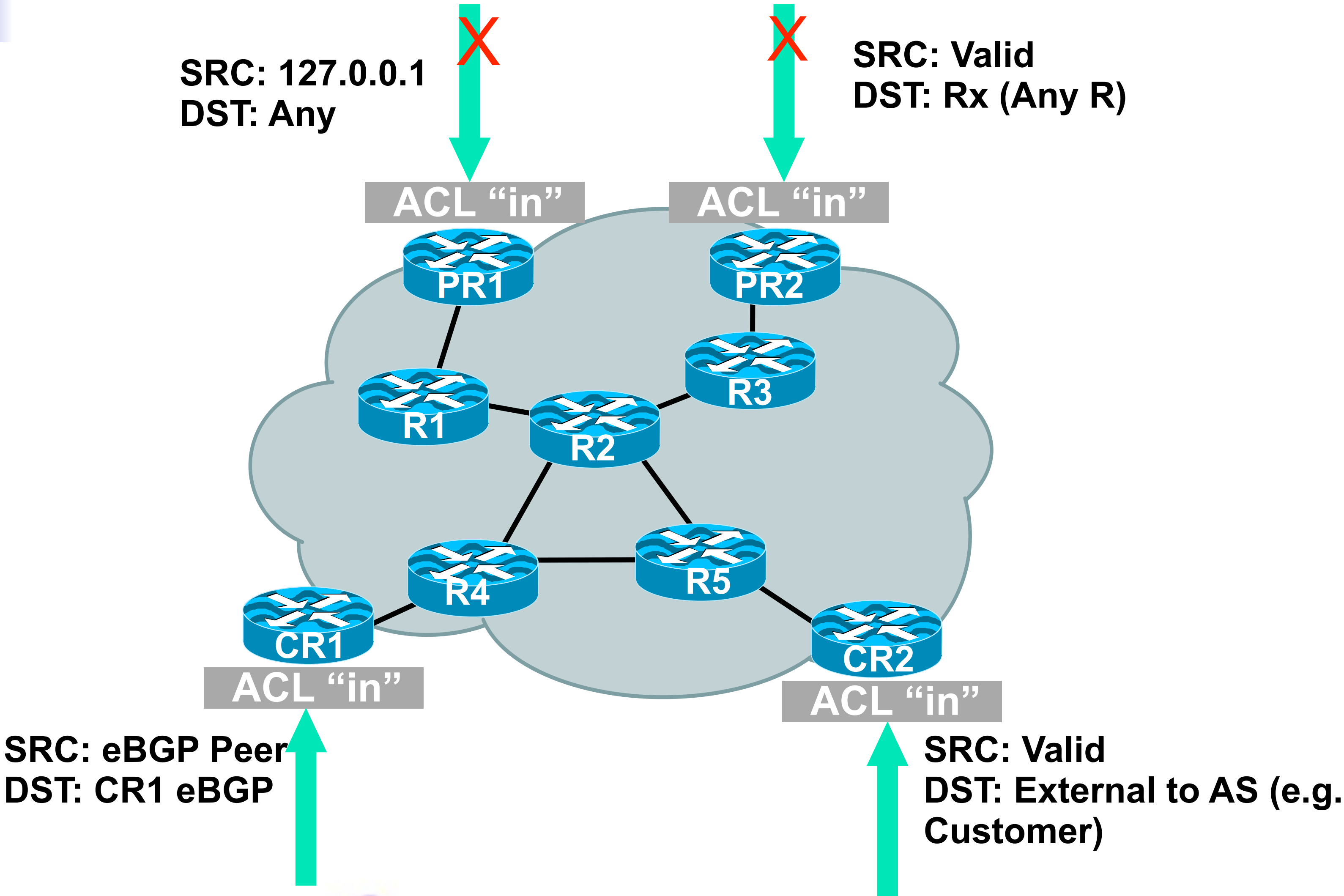
Infrastructure ACL in Action



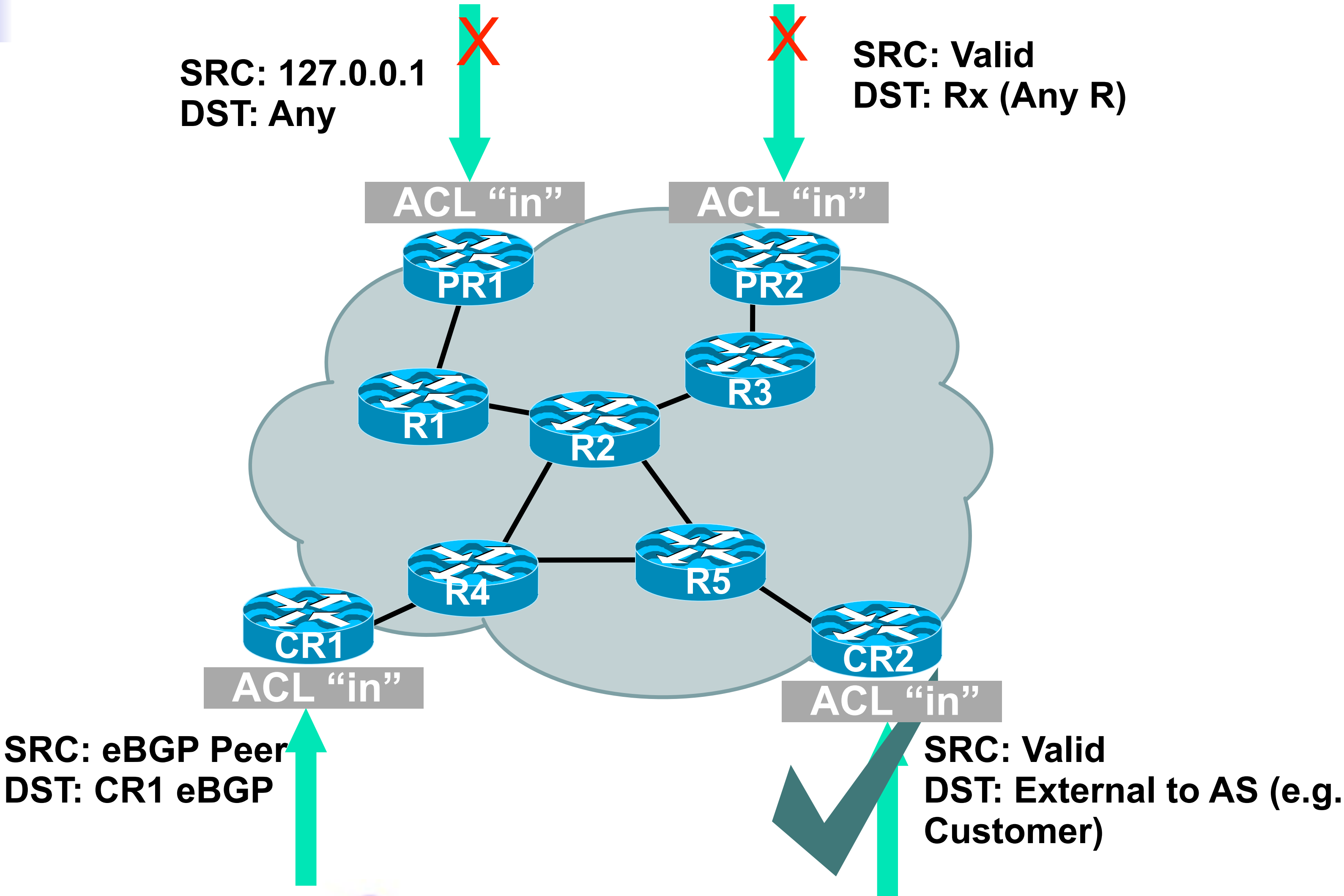
Infrastructure ACL in Action



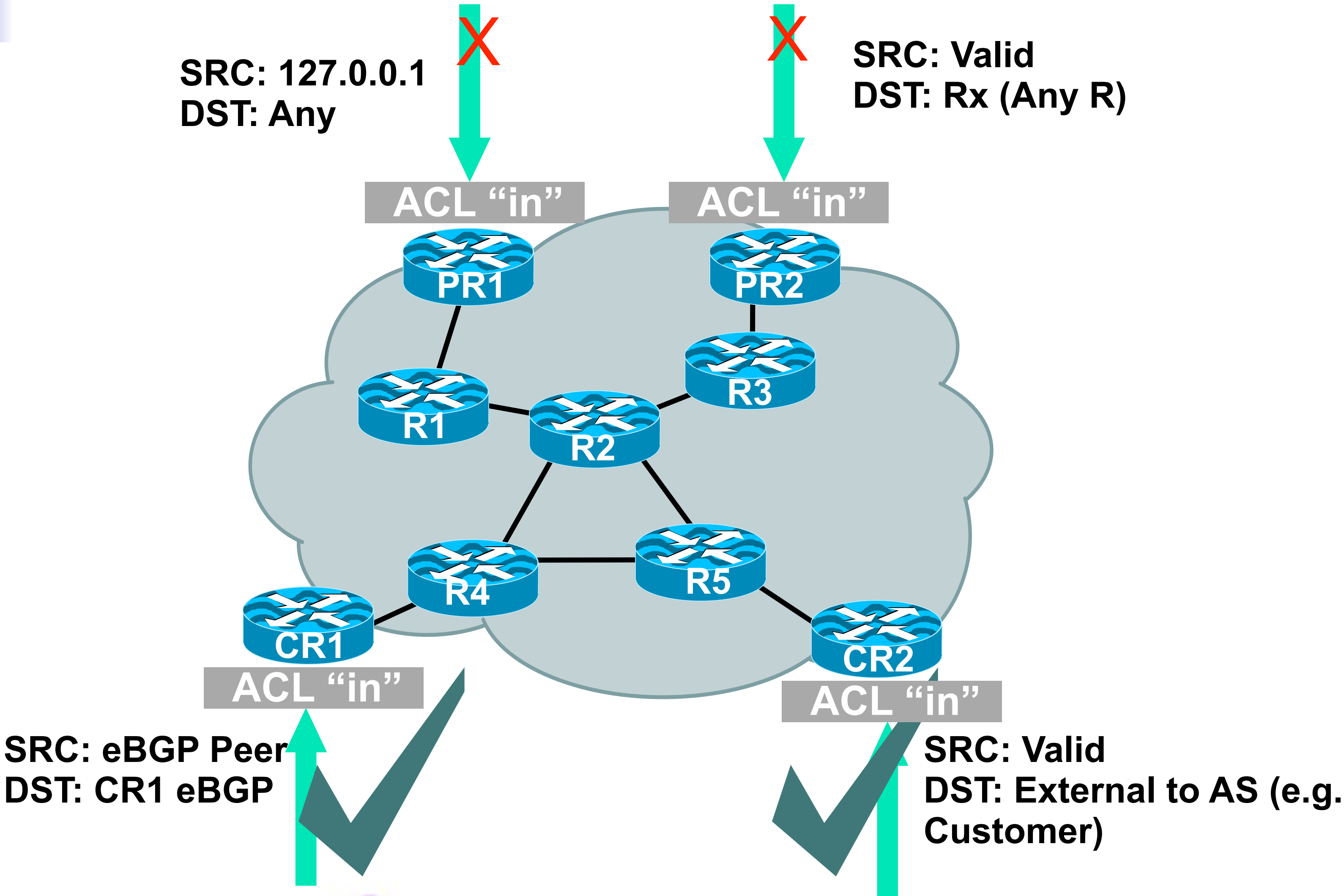
Infrastructure ACL in Action



Infrastructure ACL in Action



Infrastructure ACL in Action



IP Options

- Provide control functions that may be required in some situations but unnecessary for most common IP communications
- IP Options not switched in hardware
- Complete list and description of IP Options in RFC 791
- Drop and ignore reduce load on the route processor (RP)
- Caution: some protocols/application require options to function:
  - For example: strict/loose source routing, resource reservation protocols (RSVP) and others
- ip access-list extended drop-ip-option
  - deny   ip any any option any-options
  - permit ip any any

IP Options

- ip options drop
- ip options ignore—router ignores options
  - Best practice when router doesn't need to process options
  - "ignore" not available on all routing platforms
  - Available in 12.0(22)S, 12.3(4)T and 12.2(25)S http://www.cisco.com/en/US/products/sw/iosswrel/ps1829/products_feature_guide09186a00801d4a94.html

Iterative Deployment

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

Step 1: Classification

- Traffic destined to the core must be classified
- NetFlow can be used to classify traffic
  - Need to export and review
- Classification ACL can be used to identify required protocols
  - Series of permit statements that provide insight into required protocols
  - Initially, many protocols can be permitted, only required ones permitted in next step
  - Log keyword can be used for additional detail; hits to ACL entry with log will increase CPU utilization: impact varies by platform
- Regardless of method, unexpected results should be carefully analyzed → do not permit protocols that you can't explain!

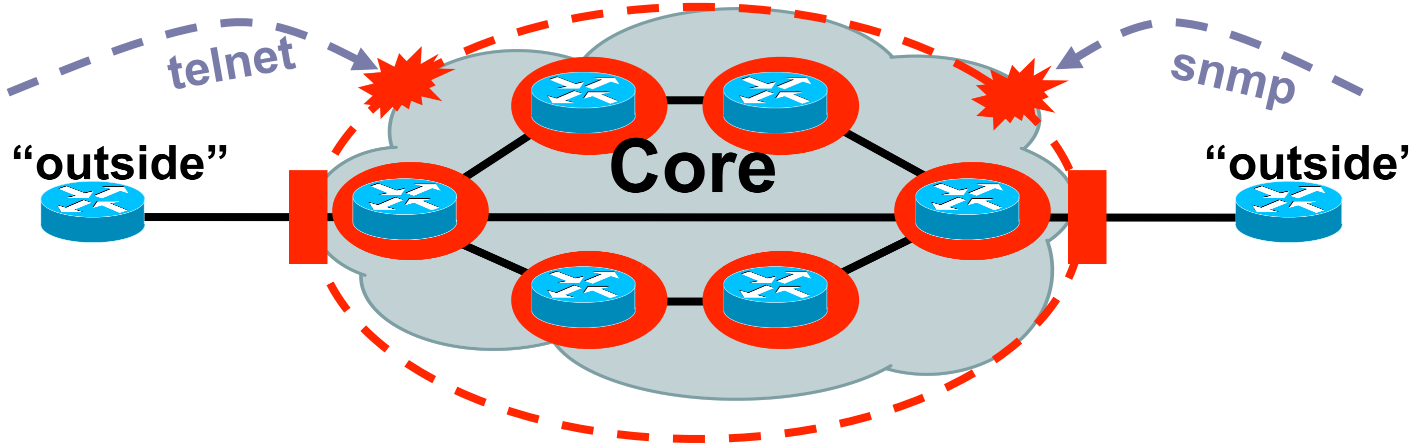
Step 2: Begin to Filter

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

Steps 3 and 4: Restrict Source Addresses

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

Infrastructure ACLs



- Edge "shield" in place
- Not perfect, but a very effective first round of defense
  - Can you apply iACLs everywhere?
  - What about packets that you cannot filter with iACLs?
  - Hardware limitations
- Next step: secure the control/management planes per box

Packet filters

Packet filter applications

- Protect control plane (RE)
- Limit services
- Drop packets that don't belong
  - Spoofed packets
  - uRPF perhaps better application

Overview of Firewall Filter Syntax

```
[edit firewall family inet]
filter filter-name {
  term term-name {
      from {
         match-conditions;
      }
     then {
       action;
       action-modifiers;
  term implicit-rule {
     then discard;
     }
   }
}
```

- Syntax similar to policy statements
- Defined under `[edit firewall family]` hierarchy level
- Named filters, one or more terms
  - Terms processed sequentially
  - All packets match a term when a from condition is not specified
  - Implicit discard all for packets that do not match any term
- Actions: accept, reject, and discard
  - Modifiers: log, count, sample, etc.
- One filter per logical unit, per direction; the same filter can be used on many interfaces

Overview of Match Conditions

- Firewall match conditions
  - Generally, each term in a filter has a match condition
    - Terms without a `from` statement match all traffic
  - The `from` statement specifies the conditions the packet must match for the action to be taken
    - Multiple match conditions possible per term
    - All conditions in the `from` statement must match (logical AND)
- Several categories of match conditions
  - Numeric-range filter
  - Address filter
  - Bit-field filter

Numeric Range Filter Match Condition

- Match packet fields that can be identified by a numeric value
  - Port and protocol numbers
- Specify a keyword that identifies the condition and a value that a field in a packet must match
  - `source-port 1024-65535`
  - `source-port smtp`
- Keywords identifying available fields:
  - `destination-port`, `dscp`, `fragment-offset`, `icmp-code`, `icmp-type`, `interface-group`, `packet-length`, `port`, `precedence`, `protocol`, `source-port`

Address Filter Match Condition

- IP source and destination prefixes
- Keywords available
  - `address prefix` (source or destination)
  - `destination-address` prefix
  - `source-address` prefix
- Address matches are longest or

Bit-Field Match Condition

- Match on specific bits in certain packet fields
- You can specify bit fields with symbolic names or numeric values
- Bit matching for IP options, fragment flags, and TCP flags
  - Note: Specification of a bit field does NOT imply the corresponding protocol
- Grouping (…), negation (!), and support for logical AND (& or +), logical OR (| or ,) functions

Bit-Field Match Examples

IP Options

loose-source-route	(131)	Strict-source-route	(137)
record-route	(7)	router-alert	(148)
Timestamp	(68)		

TCP Flags

ack (0x10)	fin (0x01)	push (0x08)	rst (0x04)
syn (0x02)	urgent (0x20)		

Example: “tcp-flags (0x01 & 0x02)” is equal to “tcp-flags 0x03”

Text Synonyms

first-fragment (matches offset = 0, MF = 1)

tcp-established: Equivalent to “(ack | rst)”

tcp-initial: Equivalent to “(syn & !ack)”

Firewall Actions Overview

- Overview
  - Actions fall into two categories
    - Actions: accept, discard, and reject
    - Action modifiers: count, sample, and log/syslog
  - Default action is discard
    - Use of an action modifier creates an implicit accept (a sampled packet automatically is accepted unless an explicit reject is included in the term)

Action Statements

- Three action statements:
  - `accept`: The packet is accepted for forwarding—no other term is analyzed
  - `reject` *message-type*: The packet is rejected, and the corresponding ICMP message is generated; no other term is analyzed
  - `discard`: The packet is silently discarded, and no other term is analyzed
    - Provides better security—very useful for DoS attacks due to address spoofing or the use of zombies

Reject Message Options

- Based on configuration, the `reject` action generates one of the following:
  - `administratively-prohibited` (default)
  - `bad-host-tos, bad-network-tos`
  - `host-prohibited`, `host-unknown`, `host-unreachable`
  - `network-prohibited`, `network-unknown`, `network-unreachable`
  - `port-unreachable`
  - `precedence-cutoff, precedence-violation`
  - `protocol-unreachable`
  - `source-host-isolated`, `source-route-failed`
  - `tcp-reset`
    - Generates a TCP reset segment when the rejected traffic is TCP;

Action Modifiers

- *counter-name*
  - The corresponding counter is incremented
    - View with `show firewall` *filter-name*
    - Clear with `clear firewall [all |` *counter-name* `|` *filter-name*`]`
- Logging
  - The packet sample is logged to the routing daemon cache
    - View with `show firewall log`
    - No clear command—displays most recent entries first
  - Also can log to syslog with syslog action
- Sampling
  - Packets are sampled and written to a file based on sampling settings
  - Specified under the `forwarding-options` hierarchy
  - Local ASCII files and cflowd version 5|8 export

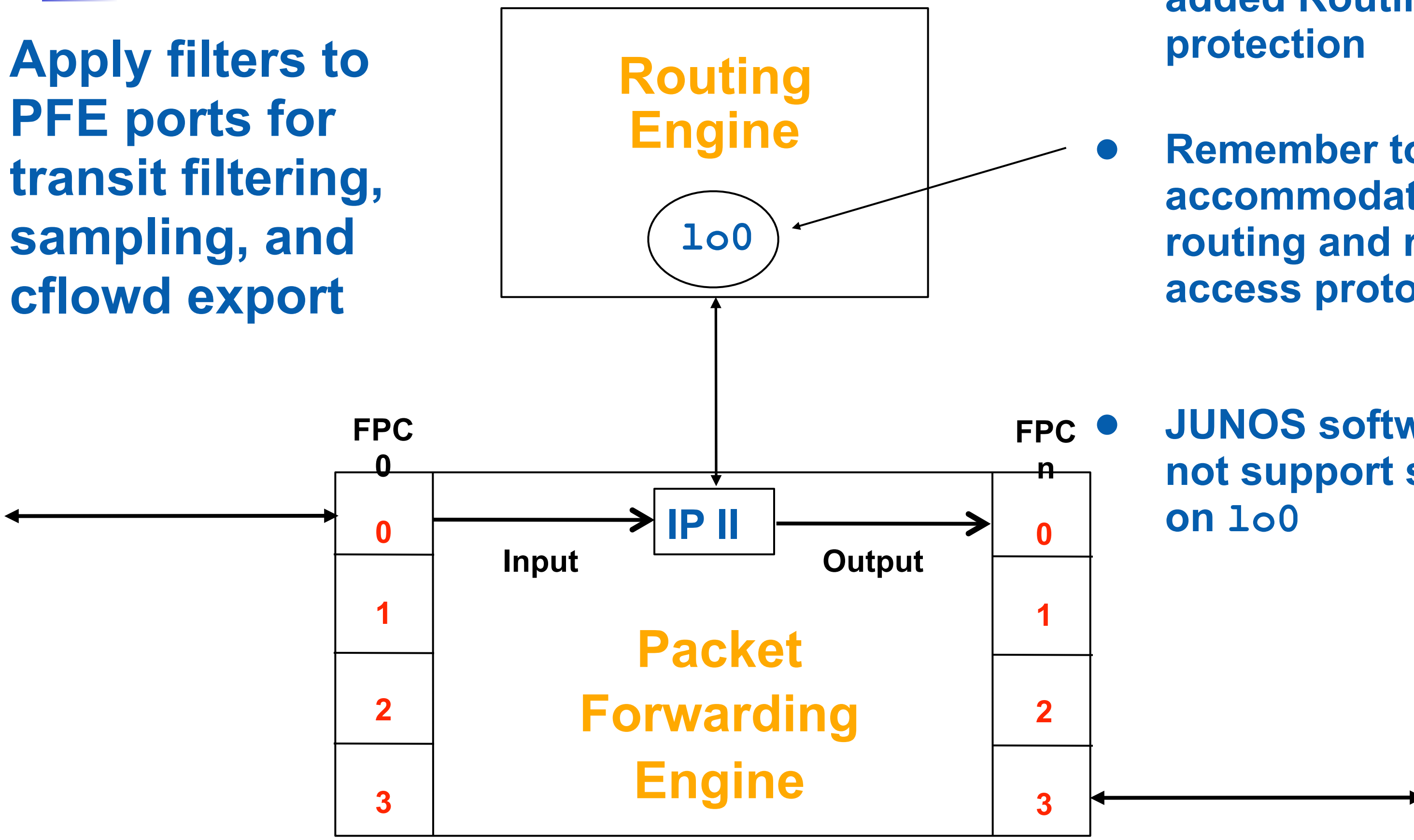
Applying Firewall Filters

```
interfaces {
   interface-name {
      unit logical-unit-number {
        family inet {
           filter {
              input filter-name;
              output filter-name;
              }
           }
      }
}
```

- Filters must be applied to an interface to take effect
- A common filter can be applied to multiple (or even all) interfaces
- Each interface can support two filters per logical unit—one input and one output
- Apply the filter to the loopback interface for Routing Engine protection

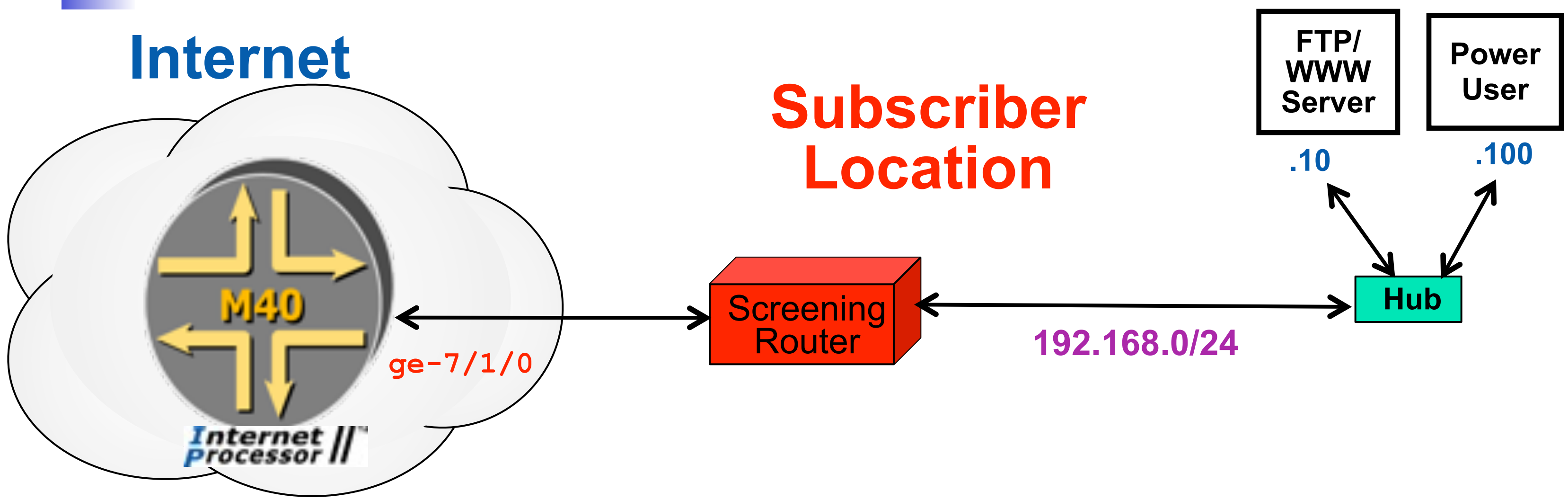
Transit versus Routing Engine Filters

Apply filters to PFE ports for transit filtering, sampling, and cflowd export



- Apply filters to `lo0` for added Routing Engine protection
- Remember to accommodate your routing and remote access protocols!
- JUNOS software does not support sampling on `lo0`

Spoof Prevention



Rule 1: Input
From SA = 192.168.0
Then Accept
From SA = other
Then Reject

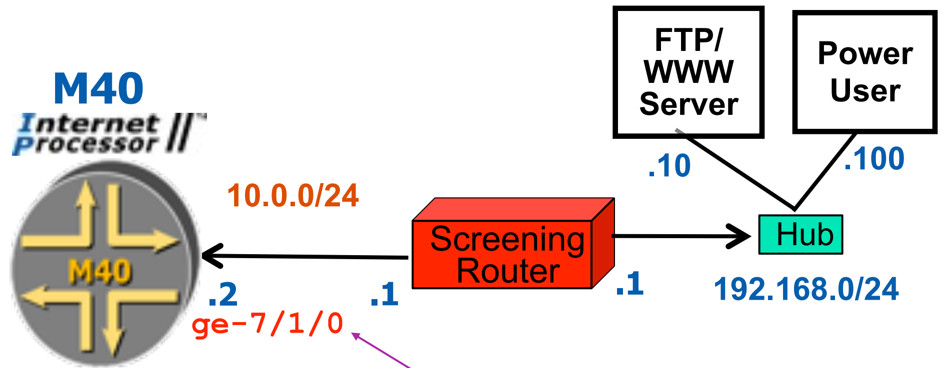
Rule 1 prevents the origination of spoofed packets from this site

Rule 2: Output
From SA = 192.168.0
Then Reject
From SA = other
Then Accept

Rule 2 blocks spoofed packets from entering this site

Inbound Spoof Prevention

```
[edit firewall family inet]
lab@router# show
filter no-spoofs-in {
    term allow-valid {
        from {
            source-address {
                192.168.0.0/24;
                10.0.0.0/24;
            }
        }
        then accept;
    }
    term reject {
        then {
            count bad-source-address;
            log;
            discard;
        }
    }
}
```

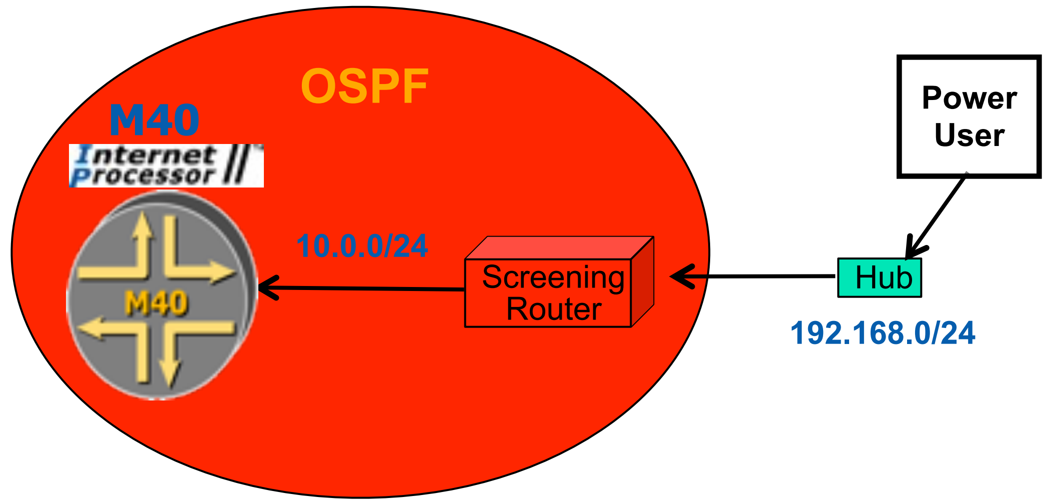


Applied as input filter on subscriber interface

```
[edit interfaces ge-7/1/0]
lab@router# show
unit 0 {
    family inet {
        filter {
            input no-spoofs-in;
        }
        address 10.0.0.2/24;
    }
}
```

Pop Quiz!

```
[edit firewall]
lab@router# show
family inet {
    filter pop-me {
        term telnet {
            from {
                protocol tcp;
                port telnet;
            }
            then accept;
        }
        term ping {
            from {
                protocol icmp;
            }
            then accept;
        }
    }
}
```



Shortly after applying this filter to the `lo0` interface, the user's Telnet session *hangs* and cannot be reestablished

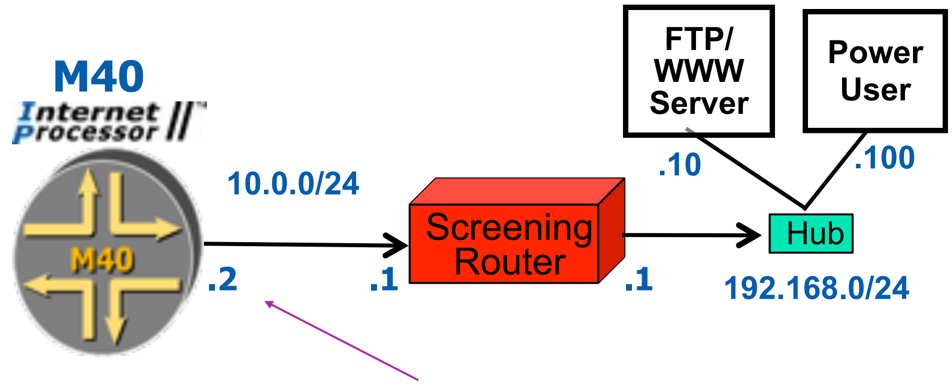
Any ideas?

Preventing Fragmentation Exploits

```
[edit firewall]
lab@San_Jose-3# show
family inet {
    filter no-frags {
        term 1 {
            from {
                is-fragment;
                protocol [ icmp udp ];
            }
            then {
                count no-frags;
                log;
                discard;
            }
        }
        term 2 {
            then accept;
        }
    }
}
```

Permits diagnostic pings while blocking fragmented ICMP/UDP traffic

(For example, Teardrop, Boink, POD)

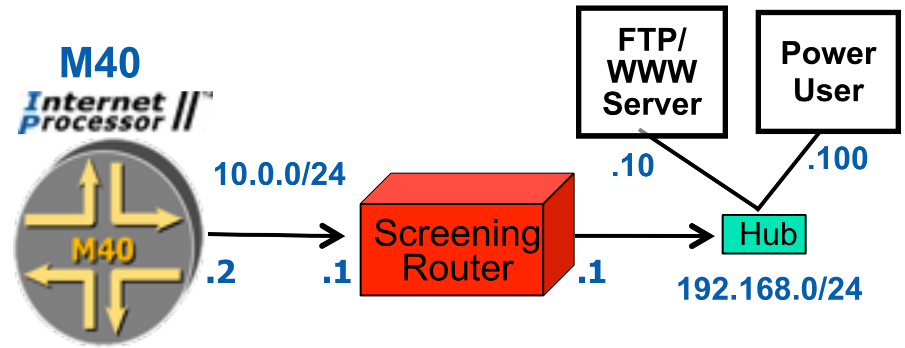


Filter applied in output direction of subscriber interface

```
[edit interfaces ge-7/1/0]
lab@router# show
unit 0 {
    family inet {
        filter {
            output no-frags;
        }
        address 10.0.0.2/24;
    }
}
```

Securing the FTP/WWW Server

```
[edit firewall family inet filter ftp-www-only]
lab@San_Jose-3# show
term allow-ftp-www {
    from {
        destination-address {
            192.168.0.10/32;
        }
        protocol tcp;
        destination-port [ ftp ftp-data http ];
    }
    then accept;
}
term reject-other {
    from {
        destination-address {
            192.168.0.10/32;
        }
    }
    then {
        count unauthorized-service-requests;
        log;
        discard;
    }
}
term accept-all {
    then accept;
}
```




```
interfaces ge-7-1/0 {
    unit 0 {
        family inet {
            filter {
                output ftp-www-only;
            }
        }
    }
}
```

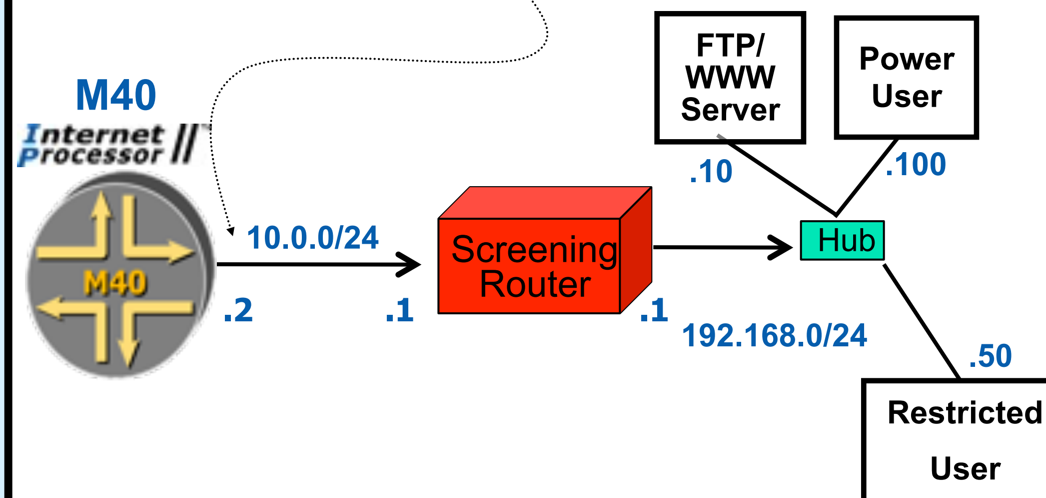
Filter applied in output direction of subscriber interface

Remember the implicit deny all for unmatched traffic!

Outgoing Service Restriction

```
[edit firewall family inet]
lab@San_Jose-3# show filter user-control
term normal-user-allow {
    from {
        destination-address {
            0.0.0.0/0;
            192.168.0.100/32 except;
        }
        protocol tcp;
        source-port http;
        tcp-established;
    }
    then accept;
}
term track-unauthorized {
    from {
        destination-address {
            0.0.0.0/0;
            192.168.0.100/32 except;
        }
    }
    then {
        count unauthorized;
        discard;
    }
}
term power-user {
    then accept;
}
```

Filter applied in output direction to filter response traffic!



Note the use of `except`, which exempts the power user from a particular terms in this filter

Rate Policing

- Instead of allowing or dropping packets that meet match conditions, you can use a filter to identify traffic that is to be policed (rate-limited)
  - You can apply a policer directly to an interface to rate-limit all traffic associated with that protocol family
- Traffic that matches the filter is then policed according to an average bandwidth and a burst size
  - Can specify bandwidth as a percentage of interface speed
- When traffic exceeds the policing parameters, it can:
  - Be discarded
  - Have its loss-priority (PLP) bit set
  - Be associated with a forwarding class (output queue)

Rate Policing Example

```
[edit firewall]
lab@router# show
policer p1 {
    if-exceeding {
        bandwidth-limit 400k;
        burst-size-limit 100k;
    }
    then discard;
}
family inet {
    filter limit-ftp {
        term ftp {
            from {
                source-address {
                    1.2.3.0/24;
                }
                protocol tcp;
                destination-port [ ftp ftp-data ];
            }
            then {
                policer p1;
                count count-ftp;
            }
        }
    }
}
```

- **Example:**
  - `bandwidth-limit`
    - **In bits per second**
    - **30,520 bps to 4.29 Gbps**
  - `burst-size-limit`
    - **In bytes per second**
    - **Min should = 10 times MTU (low speed) or bandwidth times 3–5 milliseconds (high speed)**
    - **Max = 16.7 Mb**

Interface-Based Policers

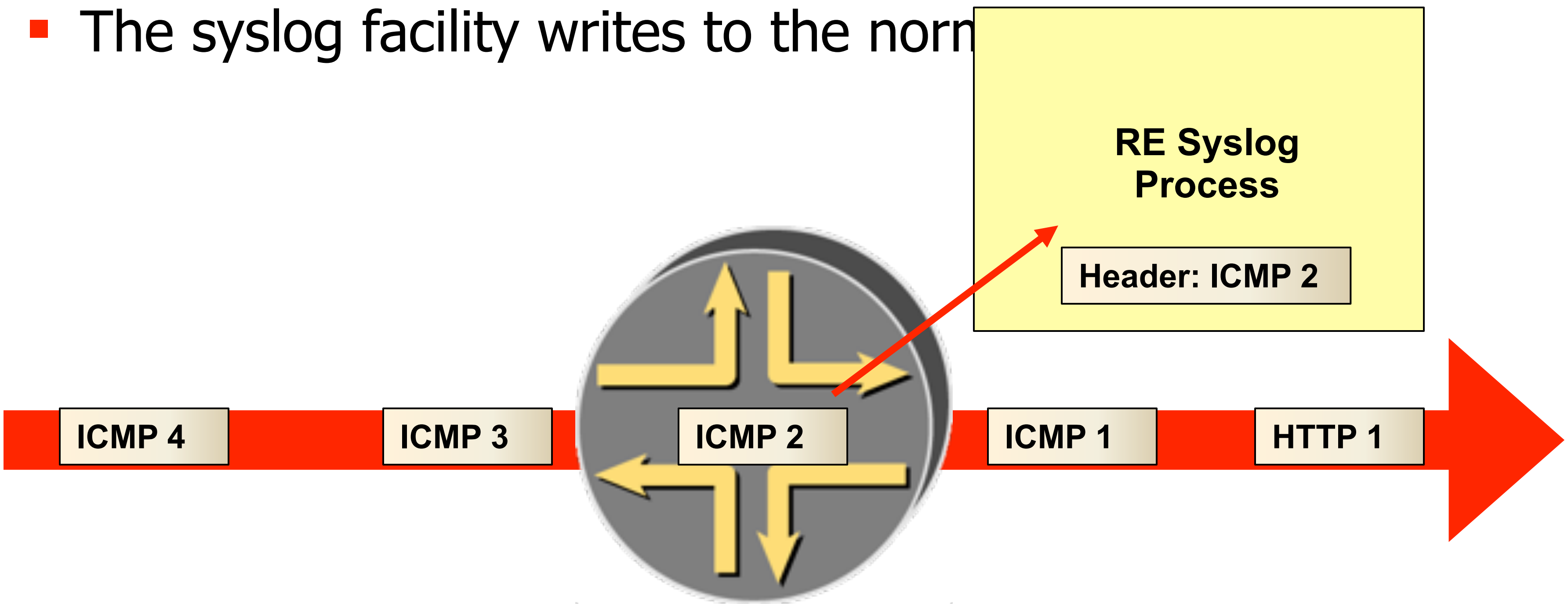
```
interfaces {
   interface-name {
       unit logical-unit-number {
             family inet {
                   filter {
                         input filter-name;
                         output filter-name;
                   }
                   policer {
                         input policer-template;
                         output policer-template;
                   }
             }
       }
}
```

Firewall-Related Operational Commands

- List of commands:
  - `show firewall name`
    - Displays counter values
  - `show firewall log`
    - Displays kernel log cache
  - `show log log-file-name`
    - Displays logged entries when the syslog action modifier is used in a term
  - `clear firewall name`
    - Resets counters associated with a firewall
  - `show policer`
    - Displays a list of interface policers
  - `show interfaces policer interface-name`
    - Displays details about interface policers

Using the Firewall `syslog` Modifier

- Sending alerts to the syslog
  - The `syslog` modifier captures minimal IP information, but allows automated detection and an audit trail
  - The syslog facility writes to the norm



Sample Filter Using Syslog

- Logging filter example:
  - Create term to syslog TCP packets with the SYN bit set with a `syslog` action modifier
  - Remember to create a final

```
[edit firewall family inet]
lab@R1# show
filter filter-test {
    term count-syn {
        from {
            protocol tcp;
            tcp-initial;
        }
        then {
            syslog;
            accept;
        }
    term others {
        then {
            count other-packets;
            accept;
        }
```

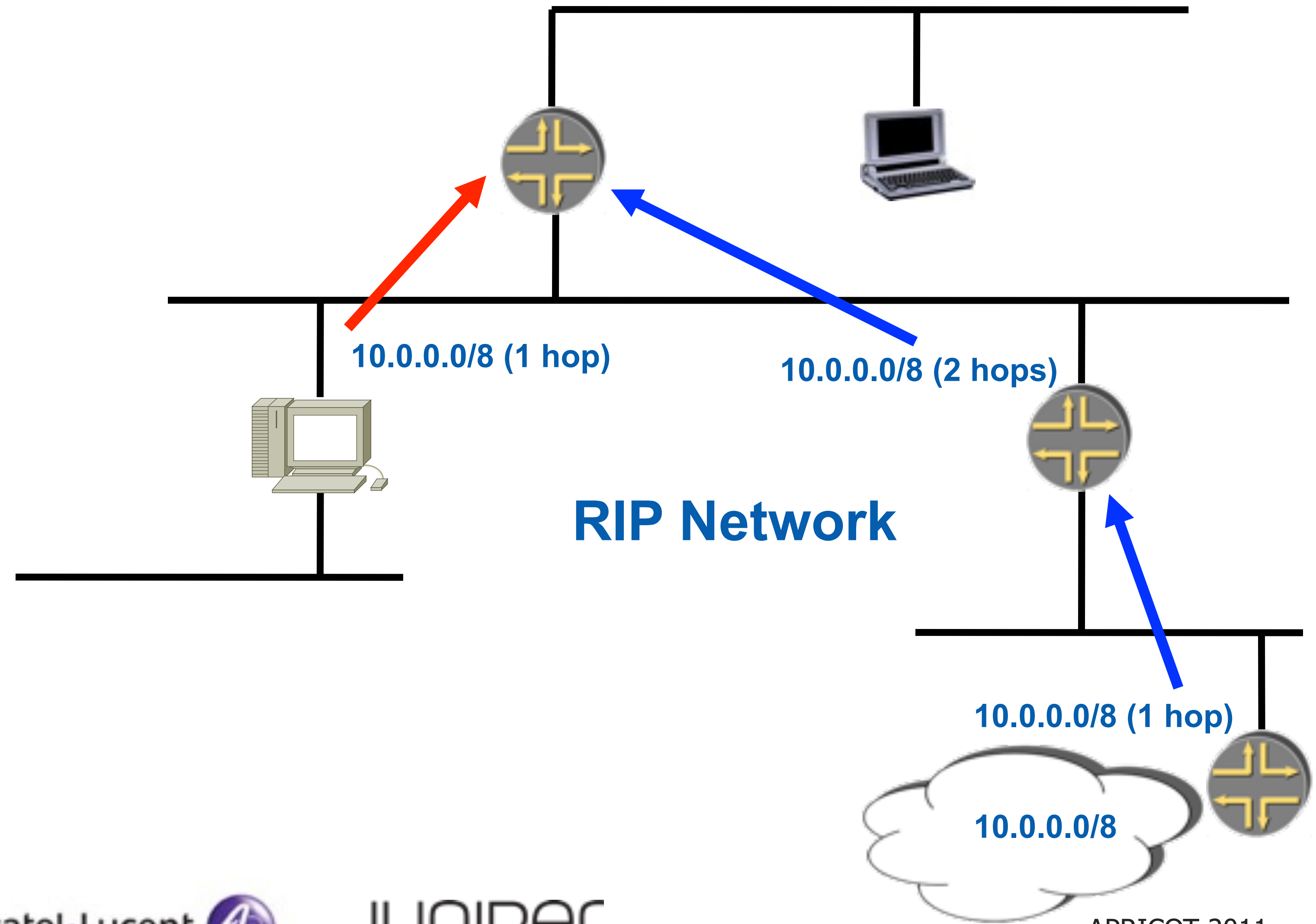
Lab: Secure the control plane

- Follow the lab guide

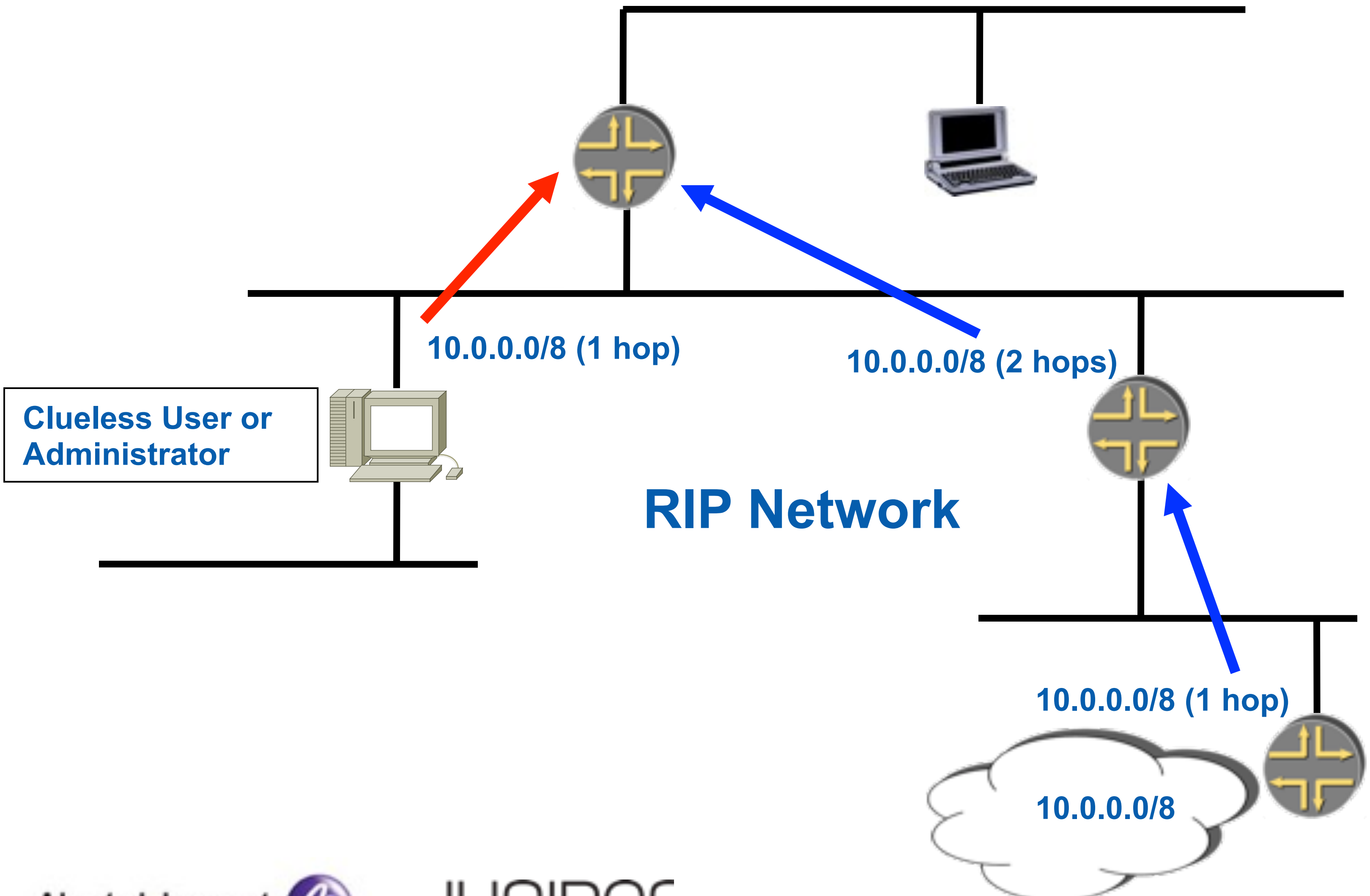
The Need for Secure Routing

- Q: What is the most important function performed by routers?
  - A: They route!

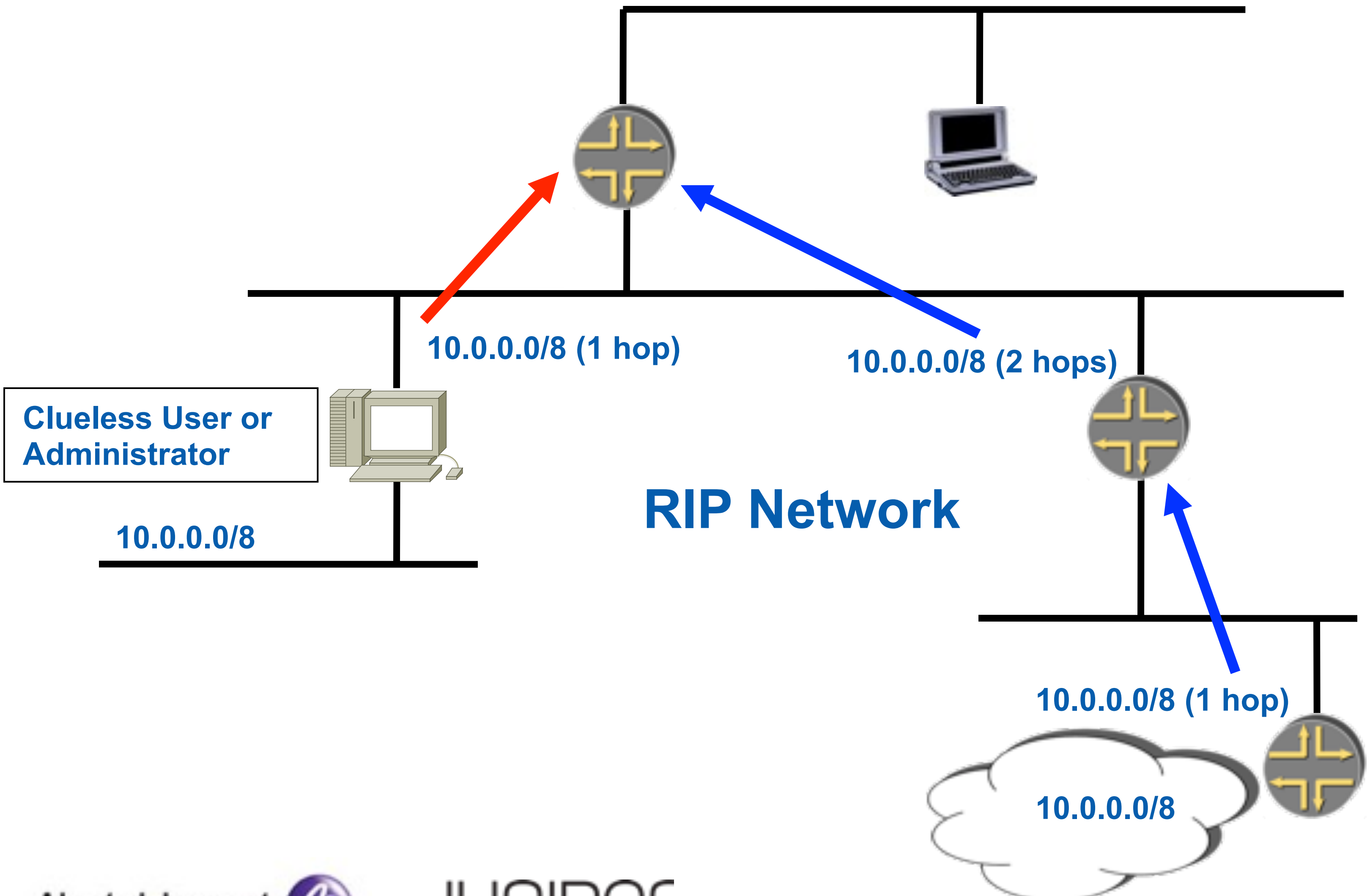
Routing Gone Bad—Example 1



Routing Gone Bad—Example 1

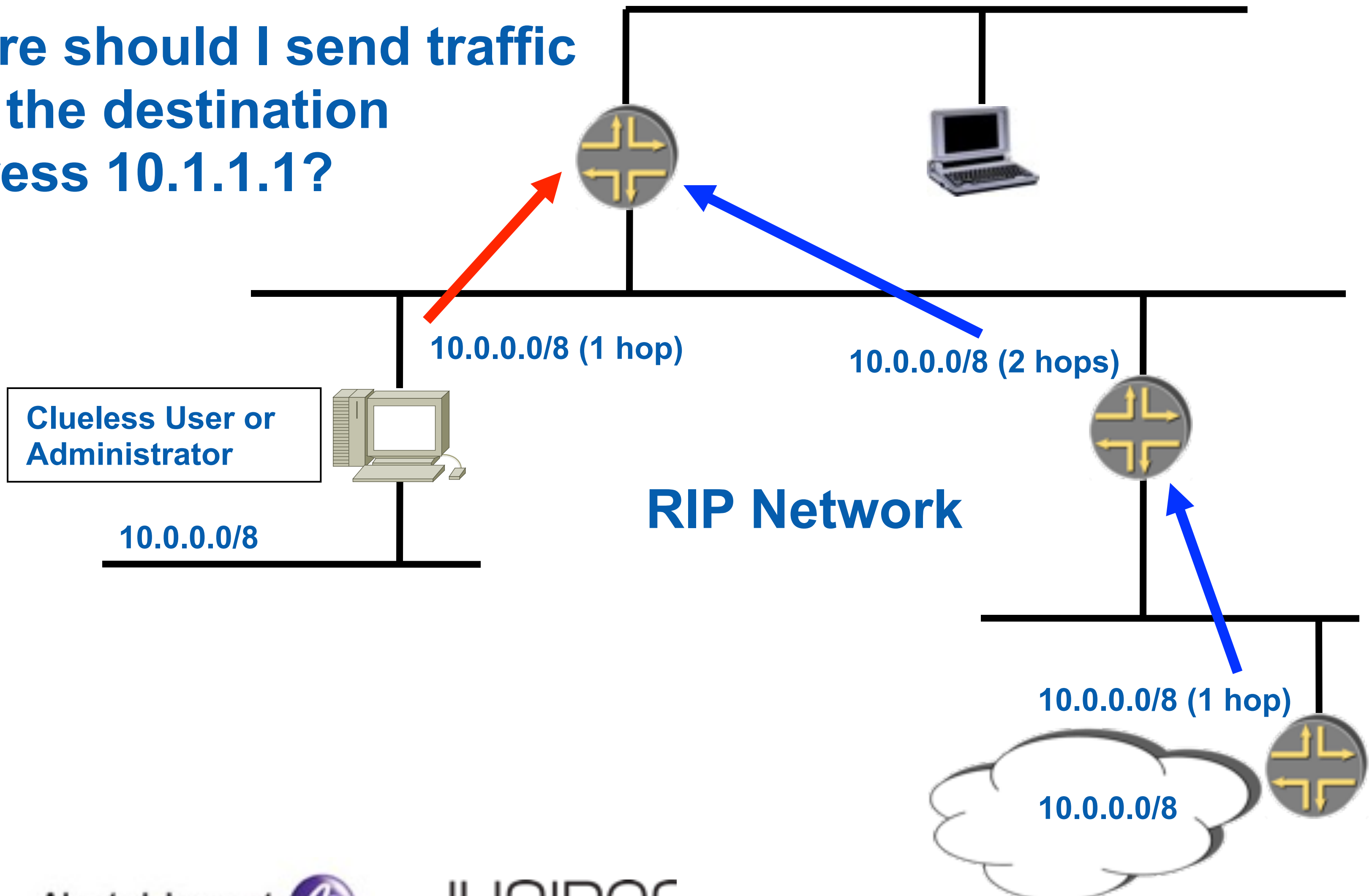


Routing Gone Bad—Example 1



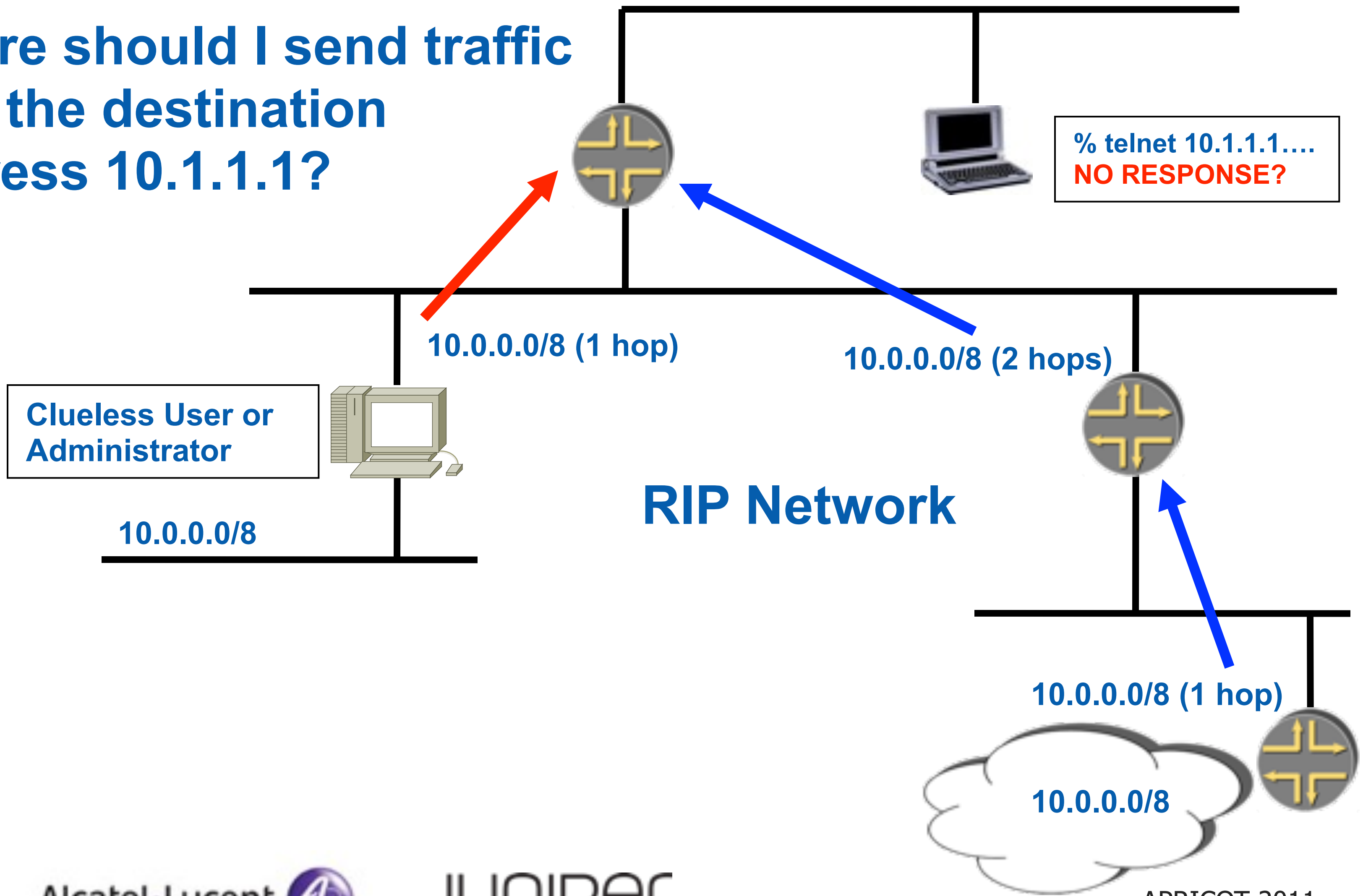
Routing Gone Bad—Example 1

Where should I send traffic with the destination address 10.1.1.1?

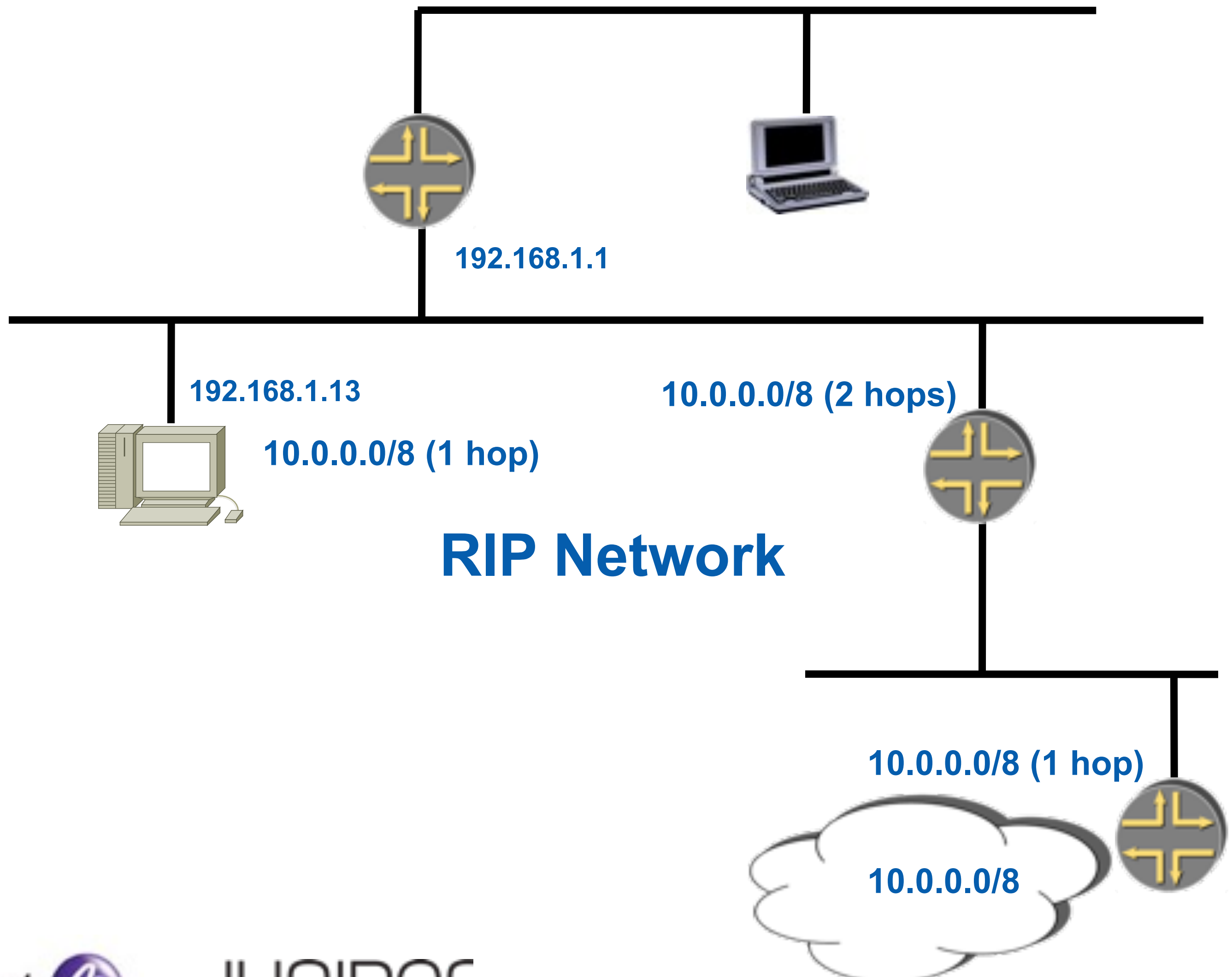


Routing Gone Bad—Example 1

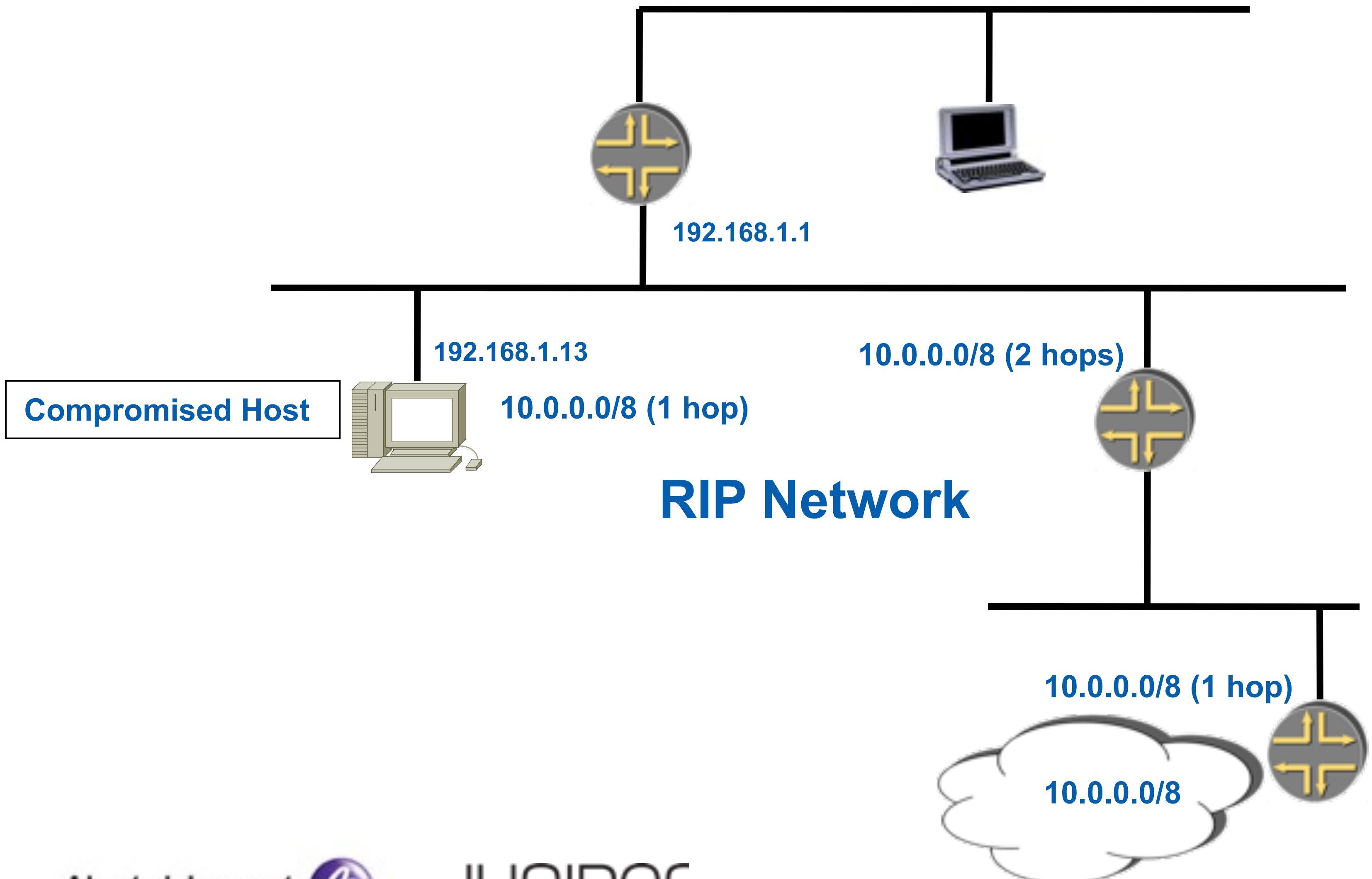
Where should I send traffic with the destination address 10.1.1.1?



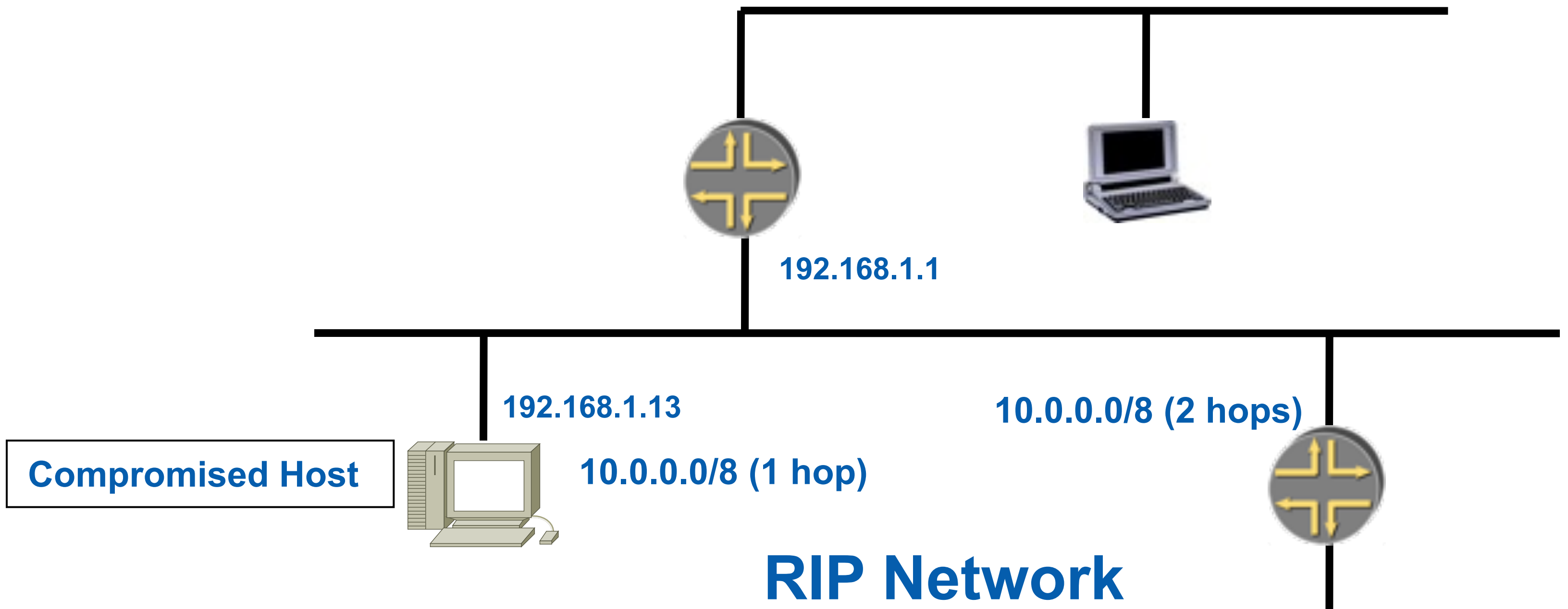
Routing Gone Worse—Example 2

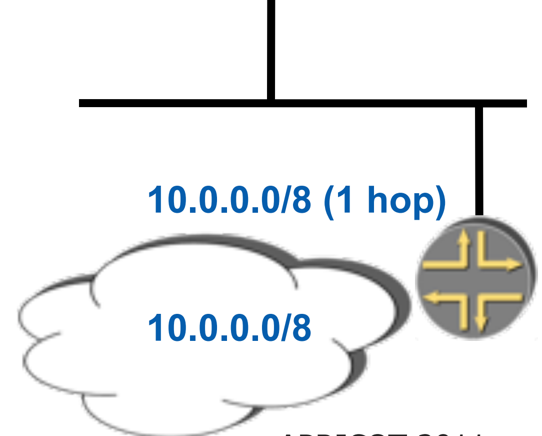


Routing Gone Worse—Example 2



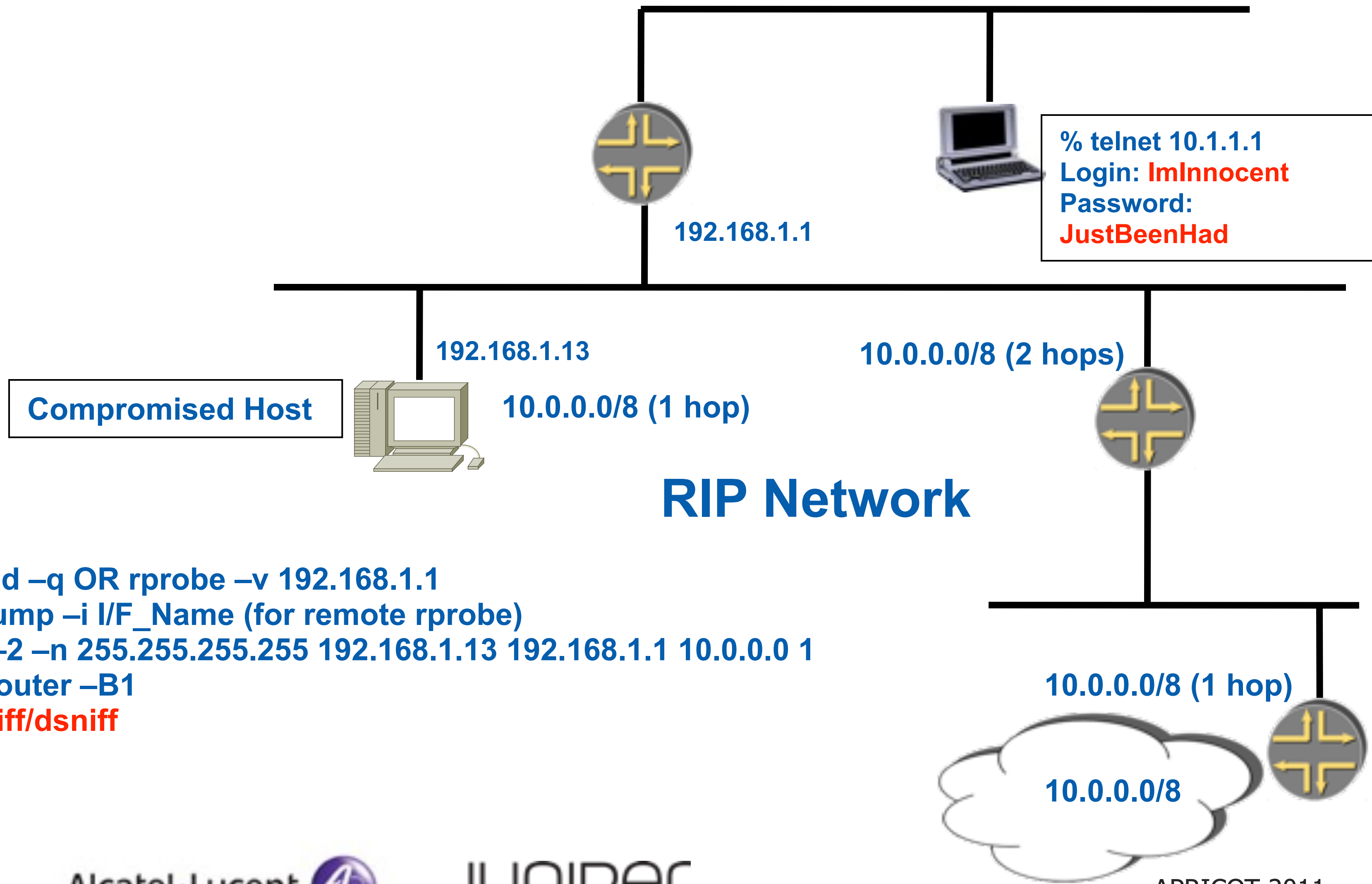
Routing Gone Worse—Example 2





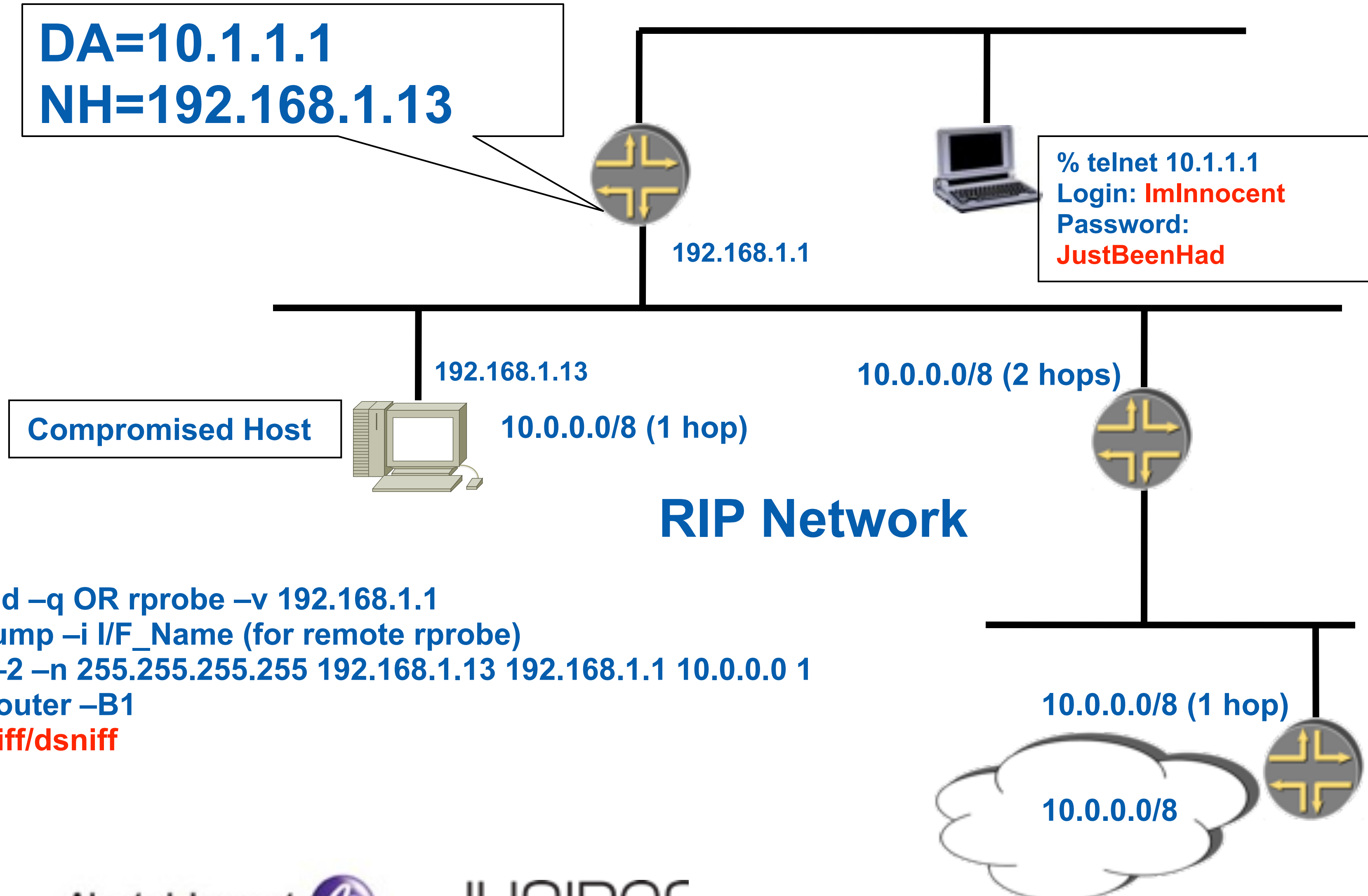
1.) routed –q OR rprobe –v 192.168.1.1
2.) tcpdump –i I/F_Name (for remote rprobe)
3.) srip –2 –n 255.255.255.255 192.168.1.13 192.168.1.1 10.0.0.0 1
4.) fragrouter –B1
5.) linsniff/dsniff

Routing Gone Worse—Example 2



1.) routed –q OR rprobe –v 192.168.1.1
2.) tcpdump –i I/F_Name (for remote rprobe)
3.) srip –2 –n 255.255.255.255 192.168.1.13 192.168.1.1 10.0.0.0 1
4.) fragrouter –B1
5.) linsniff/dsniff

Routing Gone Worse—Example 2



1.) routed –q OR rprobe –v 192.168.1.1
2.) tcpdump –i I/F_Name (for remote rprobe)
3.) srip –2 –n 255.255.255.255 192.168.1.13 192.168.1.1 10.0.0.0 1
4.) fragrouter –B1
5.) linsniff/dsniff

Routing Protocol Security

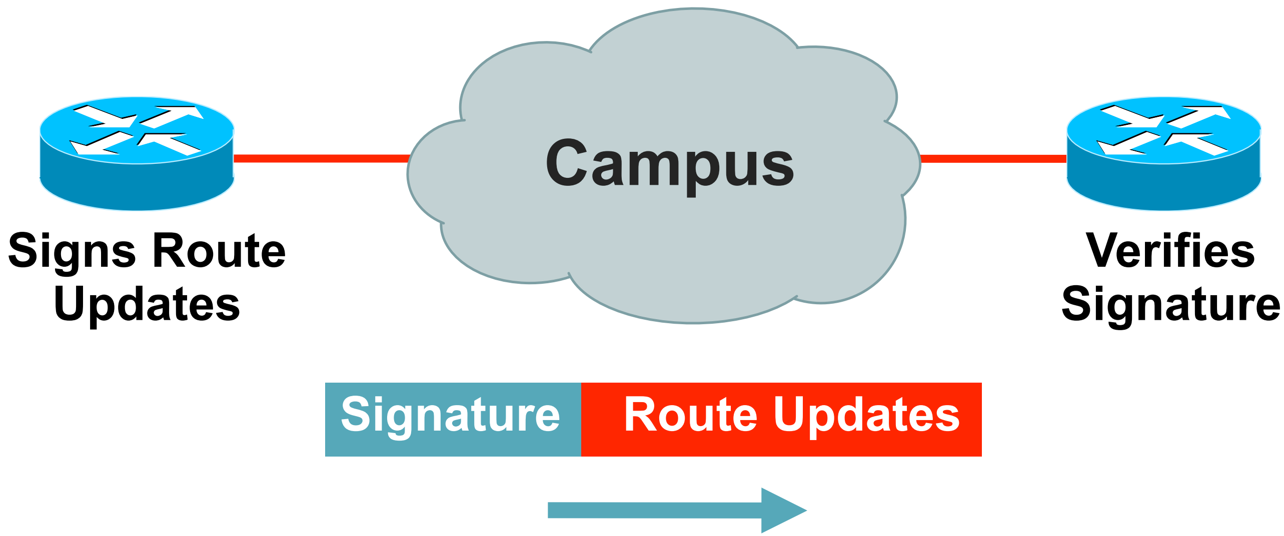
Routing Protocols Can Be Attacked

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

May Be Accidental or Intentional

Secure Routing—Route Authentication

Configure Routing Authentication



Certifies Authenticity of Neighbor and Integrity of Route Updates

Route Authentication

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

Routing Protocol Security

- The case for using routing protocol authentication:
  - RIP is not the only protocol susceptible to this attack
    - OSPF, IS-IS, and BGP are all trusting by nature
  - Implementations of routing protocols are widely available for many operating systems
    - Easy to install on a compromised host
    - Easy to implement by a disgruntled employee
    - Easy to install by a well-meaning—but clueless—insider
  - Secure your routing protocols such that only authenticated neighbors and peers can participate
  - Limit your usage of routing protocols to only the links that must be included

Agenda: Securing RIP

- Routing Protocol Authentication
- → Securing RIP
- Securing OSPF
- Securing IS-IS
- Securing BGP
- Additional Routing Security

RIP Authentication

- RIP authentication only available when using RIPv2
  - Three types of authentication are available:
    - None (default)
    - Simple (clear text)
    - MD5
  - Using MD5 authentication, a trailer is added to each RIPv2 packet
    - Contains a keyed MD5 hash of the packet contents and a shared key
    - Provides integrity, but not availability or confidentiality
  - Two places to configure RIPv2 authentication:
    - Globally, for all RIPv2 neighbors (interfaces)
    - On an individual interface (neighbor level)

RIP Configuration (Global)

```
[edit protocols rip]
lab@R1# show
authentication-type md5;
authentication-key "$9$fQ6AEhrlvL1RhrvM-dqmfz9pKvL7dsO1";
# SECRET-DATA
group inside {
    neighbor fe-0/0/1.0;
}
group outside {
    neighbor fe-0/0/0.0;
}
```

In this example, MD5 authentication is enabled for all interfaces (neighbors) running RIPv2

RIP Configuration (Neighbor)

```
[edit protocols rip]
lab@R1# show
group inside {
    neighbor fe-0/0/1.0 {
        authentication-type md5;
        authentication-key "$9$qPz6B1hcrvuO1hrlXxjHqf39yrv8xdtu"; # SECRET-DATA
    }
}
group outside {
    neighbor fe-0/0/0.0;
}
```

In this example, MD5 authentication is enabled for a single interface running RIPv2

Agenda: Securing OSPF

- Routing Protocol Authentication
- Securing RIP
- → Securing OSPF
- Securing IS-IS
- Securing BGP
- Additional Routing Security

OSPF Authentication, Configuration

- Authentication occurs within an individual area
  - Three types are supported: none, simple, and MD5
- Each interface requires an authentication key
  - Multiple interfaces can use the same key
  - Keys are always encrypted in the configuration
- By default, the authentication type is set to none
  - Effectively means no authentication is performed
- Type simple uses a plain-text password

```
[edit protocols ospf]
user@host# show
area 0.0.0.2 {
    authentication-type simple;
    interface ge-0/0/0.0 {
        authentication-key "$9$-TbwgPfzn6A";
    }
}
```

MD5 Authentication Configuration

- Using MD5 authentication, a message digest is generated and appended to the end of each OSPF packet
  - Contains a keyed MD5 hash of the packet contents and a shared key
  - Provides integrity, but not availability or confidentiality
- Each interface requires an authentication key
  - Multiple interfaces can use the same key
  - Keys are always encrypted in the configuration
- Each key requires a key ID value ranging from 0 to 255
  - If omitted, a value of 0 is used

```
[edit protocols ospf]
user@host# show
area 0.0.0.1 {
    authentication-type md5;
    interface so-0/3/1.0 {
        authentication-key "$9$ul8b0IcyrvL7VKM" key-id 10;
    }
}
```

Agenda: Securing IS-IS

- Routing Protocol Authentication
- Securing RIP
- Securing OSPF
- → Securing IS-IS
- Securing BGP
- Additional Routing Security

IS-IS Authentication

- Authentication can occur within multiple places
  - Level 1
  - Level 2
  - Interface
- Three authentication types are supported
  - None (default)
  - Simple
  - MD5
- Using HMAC-MD5 authentication, TLV 10 is included in each IS-IS PDU
  - TLV contains an HMAC-MD5 hash of the packet contents and a shared key
  - Provides integrity, but not availability or confidentiality

Authentication Configuration

- Level authentication affects all IS-IS PDUs
  - Link-state, sequence number, and hello
- Per-interface authentication takes precedence over per-level settings

```
[edit protocols isis]
user@host# show
level 1 {
    authentication-key "$9$bssYomPQ69pkq39puhc8X7V2a"; # SECRET-DATA
    authentication-type md5;
}
level 2 {
    authentication-key "$9$dXVYoDjqQ39gomTz6CAvW8X-ViHmFnCDi1h"; # SECRET-DATA
    authentication-type simple;
}
interface fe-0/0/0.0 {
    level 2 {
        hello-authentication-key "$9$1sEEclw4JH-d2oGq.Ctp01h7NbgaU"; # SECRET-DATA
        hello-authentication-type md5;
}
interface so-0/1/0.0;
```

Authentication Issues

- Hello authentication only secures IS-IS hello packets
  - Determines whether an adjacency forms between two routers
- Level 1 or Level 2 authentication can be disabled for specific PDUs
  - LSP packets
  - CSNP packets (`no-csnp-authentication`)
  - PSP packets (`no-psnp-authentication`)
  - IS-IS hello packets (`no-hello-authentication`)
- Authentication for LSPs allows other routers to read the TLV values and use that information in the SPF calculation
- Disables the authentication check with the `no-authentication-check` command
  - Useful for migration purposes

OSPF and ISIS Authentication Example

OSPF

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

ISIS

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

Agenda: Securing BGP

- Routing Protocol Authentication
- Securing RIP
- Securing OSPF
- Securing IS-IS
- → Securing BGP
- Additional Routing Security

BGP Authentication

- BGP authentication:
  - Two types of authentication available
    - None (default)
    - MD5
  - Using MD5 authentication, an extension is included in each TCP segment
    - Contains a 16-byte MD5 hash of the packet contents and a shared key
    - Provides integrity, but not availability or confidentiality
- Three places to apply BGP authentication:
  - Globally for all peers
  - For all peers in an individual group
  - For a single peer

Configuration—Global, Group, or Peer

```
[edit protocols bgp]
lab@R1# show
authentication-key "$9$9tiTtpByrvMLNhSrvLXwsfTz"; # SECRET-DATA
group external {
     type external;
     neighbor 172.16.1.2 {
     authentication-key "$9$JAUDkQz6/A0fTz6AtREVwY"; # SECRET-DATA
          peer-as 64513;
     }
     neighbor 172.16.1.3 {
          peer-as 64512;
     }
}
group internal {
     type internal;
     local-address 10.1.255.1;
     authentication-key "$9$.f5FtpB1Ey9ApBEhvMJGD"; # SECRET-DATA
```

BGP Route Authentication

```
router bgp 200
 no synchronization
 neighbor 4.1.2.1 remote-as 300
 neighbor 4.1.2.1 description Link to
  Excalibur
 neighbor 4.1.2.1 send-community
 neighbor 4.1.2.1 version 4
 neighbor 4.1.2.1 soft-reconfiguration
  inbound
 neighbor 4.1.2.1 route-map Community1 out
 neighbor 4.1.2.1 password 7 q23dc%$#ert
```

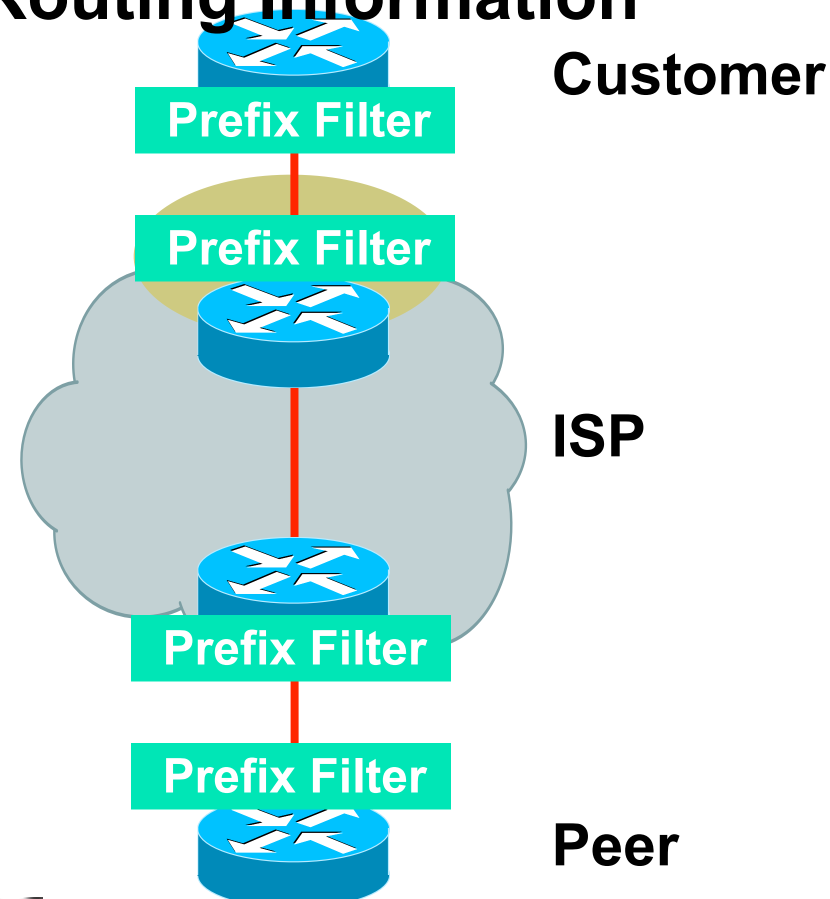
BGP Route Authentication

- Works per neighbor or for an entire peer-group
- Two routers with password mismatch:
  - %TCP-6-BADAUTH: Invalid MD5 digest from [peer's IP address]:11004 to [local router's IP address]:179
- One router has a password and the other does not:
  - %TCP-6-BADAUTH: No MD5 digest from [peer's IP address]:11003 to [local router's IP address]:179

Prefix Filters

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

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



Route Filtering

- Consider filtering routes on input to a protocol
  - You can do this on both RIP and BGP
  - The architecture of OSPF and IS-IS does not allow input filters to be placed on them
- Possible inbound filters include:
  - Reject any of your own routes advertised to you
  - Reject any external RFC 1918 routes advertised to you
  - Reject any external routes from reserved or unallocated address space
  - Reject any instance of the default route sent to you

BGP thoughts

- Consider what routes you should get from peer
  - Accept those
  - Filter all as default
- Customer (non transit) BGP routes should be received with
  - Customer AS Number, AS-Path length = 1,
  - Customer address range only
  - What degree of sub-netting will you allow
    - What is useful?

Extended ACL for a BGP Distribute List

```
access-list 150 deny ip host 0.0.0.0 any
access-list 150 deny ip 10.0.0.0 0.255.255.255 255.0.0.0 0.255.255.255
access-list 150 deny ip 127.0.0.0 0.255.255.255 255.0.0.0 0.255.255.255
access-list 150 deny ip 169.254.0.0 0.0.255.255 255.255.0.0 0.0.255.255
access-list 150 deny ip 172.16.0.0 0.15.255.255 255.240.0.0 0.15.255.255
access-list 150 deny ip 192.0.2.0 0.0.0.255 255.255.255.0 0.0.0.255
access-list 150 deny ip 192.168.0.0 0.0.255.255 255.255.0.0 0.0.255.255
access-list 150 deny ip 224.0.0.0 31.255.255.255 224.0.0.0 31.255.255.255
access-list 150 permit ip any any
```

BGP with Distribute List Route Filtering

```
router bgp 65535
no synchronization
bgp dampening
 neighbor 220.220.4.1 remote-as 210
 neighbor 220.220.4.1 version 4
 neighbor 220.220.4.1 distribute-list 150 in
 neighbor 220.220.4.1 distribute-list 150 out
 neighbor 222.222.8.1 remote-as 220
 neighbor 222.222.8.1 version 4
 neighbor 222.222.8.1 distribute-list 150 in
 neighbor 222.222.8.1 distribute-list 150 out
 no auto-summary
 !
```

Prefix-List for a BGP Prefix List

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

BGP with Prefix-List Route Filtering

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

Agenda: Additional Routing Security

- Routing Protocol Authentication
- Securing RIP
- Securing OSPF
- Securing IS-IS
- Securing BGP
- → Additional Routing Security

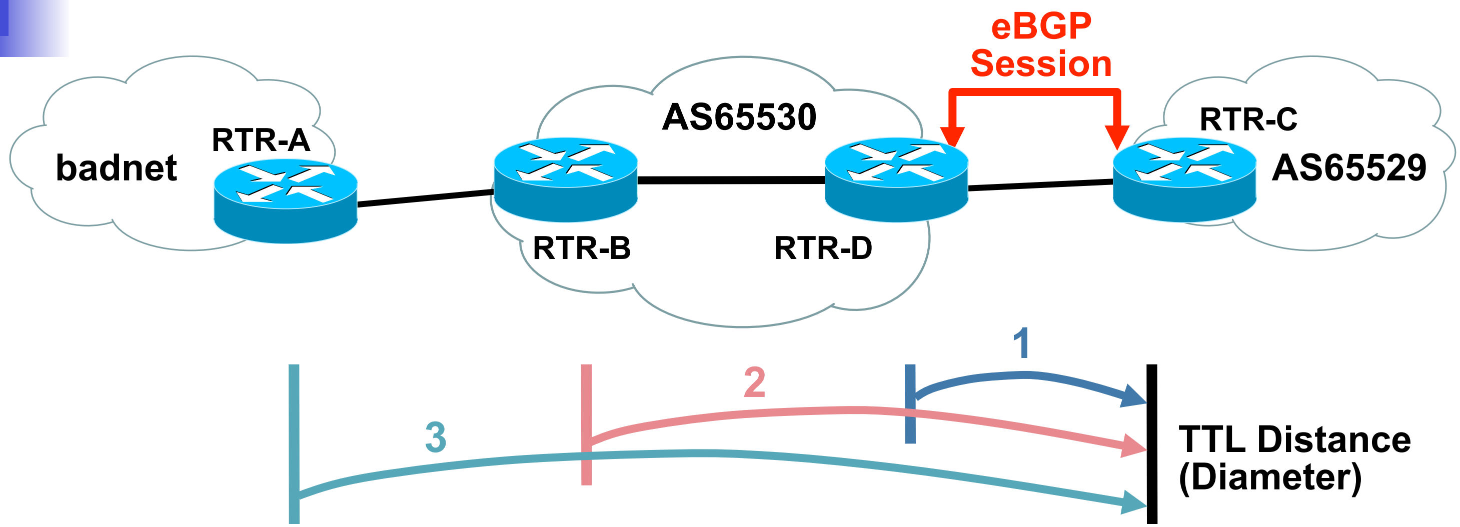
Other BGP thoughts

- Clean routes
  - Will you accept MEDS?
  - Leave communities alone
    - If you honor extended community format all should be OK
    - AS:nn
  - Is default allowed
  - Filter martians and bogons

BGP Support for TTL Security Check

- AKA BGP TTL Security Hack (BTSH)
- Protects eBGP sessions from CPU attacks using forged IP packets
- Prevents attempts to hijack eBGP session by attacker not part of either BGP network or that is not between the eBGP peers
- Configure minimum Time To Live (TTL) for incoming IP packets from a specific eBGP peer
  - BGP session established and maintained only if TTL in IP packet header is equal to or greater than configured TTL value. Initial TTL set to 255
  - If value is less than configured value packet is silently discarded and no ICMP message generated
- Not supported for iBGP and occurs after MD5 check if enabled
- Available in 12.0(27)S, 12.3(7)T, and 12.2(25)S
  - http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123newft/123t/123t_7/gt_btsh.htm

BGP TTL Security Check: How Does It Work?



Example on RTR-C:

```
router bgp 65529
neighbor 10.1.1.1 ttl-security hops 1
! expected TTL value in the IP packet header is 254
```

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

Not really security related…but…

- Extensive use of policies to provide customer services
  - E.g. Provider provisioned Local Pref
  - Examples at www.sprint.net
    - Under BGP policies

Martian Addresses

- One way to filter is to add prefixes to your martian address list
  - Address prefixes for which the routers ignore all associated routing information
- Martians are not installed into the routing table
- In JUNOS software, the default martian addresses are:
  - `0.0.0.0/8 orlonger`
  - `127.0.0.0/8 orlonger`
  - `128.0.0.0/16 orlonger`
  - `191.255.0.0/16 orlonger`
  - `192.0.0.0/24 orlonger`
  - `223.255.255.0/24 orlonger`
  - `240.0.0.0/4 orlonger`

Adding Martian Addresses

- Additional prefixes can be added to the martian list
  - This example adds all RFC 1918 addresses to the list
- Configured at the `routing-options` hierarchy level

```
routing-options {
   martians {
      destination-prefix match-type;
      }
   }

[edit]
routing-options {
            martians {
            10.0.0.0/8 orlonger;
            172.16.0.0/12 orlonger;
```

Bogons

- Addresses that are unallocated
- MUST MUST MUST MUST MUST MUST
  - Keep them up to date!
- www.cymru.com

Additional BGP Security

- IPSec transport mode between peers
- Consider enabling route damping
  - Limits the number of times an unstable (or compromised) peer can destabilize your own routing process
- Consider establishing a limit to the number of prefixes you will accept from a peer

```
[edit protocols bgp group external]
lab@R1# show
neighbor 10.1.1.1 {
    family inet {
        any {
            prefix-limit {
                maximum 125000;
                teardown 85 idle-timeout 30;
            }
}
```

IPSec—Global, Group, or Peer

```
[edit protocols bgp]
lab@R1# show
Ipsec-sa All-BGP-Neighbors
group external {
    type external;
    neighbor 172.16.1.2 {
      ipsec-sa Just-1-eBGP-Neighbor
        peer-as 64513;
    }
    neighbor 172.16.1.3 {
        peer-as 64512;
    }
}
group internal {
    type internal;
    local-address 10.1.255.1;
    ipsec-sa Just-iBGP-Neighbors
    neighbor 10.1.255.2;
    neighbor 10.1.255.3;
}
```

More BGP Security

- Since JUNOS software Release 5.4, no response to unconfigured peers
  - Negates TCP DoS attacks against TCP port 179
  - Combined with firewall filters

```
[edit policy-options]
user@R1# show | display inheritance
prefix-list ibgp-peers {
    ##
    ## apply-path was expanded to:
    ##      10.2.255.2;
    ##      10.2.255.3;
    ##
    apply-path "protocols bgp group <*> neighbor <*>";
}
```

Source MAC Address Filtering

In shared peering environments over broadcast capable media, consider using source MAC address filtering

```
[edit interfaces fe-0/0/3]
lab@R1# show
fastether-options {
    source-filtering;
    source-address-filter {
        00:e0:18:01:18:4c;
    }
}
unit 0 {
    family inet {
        address 10.2.100.1/24;
}
```

Verifying Authentication

- Authentication information available with the `show ospf interface detail` command
  - Type of authentication is displayed
  - Key ID values shown if appropriate

```
user@host> show ospf interface detail
Interface            State     Area            DR ID          BDR ID         Nbrs
fe-0/0/2.0           DR       0.0.0.0         192.168.36.1    192.168.24.1     1
Type LAN, address 10.222.4.2, mask 255.255.255.0, MTU 1500, cost 1
DR addr 10.222.4.2, BDR addr 10.222.4.1, adj count 1, priority 128
Hello 10, Dead 40, ReXmit 5, Not Stub
Auth type MD5, Active key id 4, Start time 2003 Apr 14 11:05:00 UTC
fe-0/0/3.0           DRother  0.0.0.0         0.0.0.0         0.0.0.0          0
Type LAN, address 1.1.1.2, mask 255.255.255.0, MTU 1500, cost 1
  adj count 0, priority 128
Hello 10, Dead 40, ReXmit 5, Not Stub
Auth type Password
```

BGP Attack Vectors

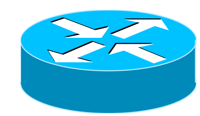
- Understanding BGP Attack Vectors will help you plan and prioritize the techniques deployed to build greater resistance into the system.
- The following documents will help you gain perspective on the realistic Risk Assessment:
  - NANOG 25 - BGP Security Update
    - http://www.nanog.org/mtg-0206/barry.html
  - NANOG 28 - BGP Vulnerability Testing: Separating Fact from FUD
    - http://www.nanog.org/mtg-0306/franz.html
- Look for the updates links to get the latest risk assessments.
  - http://www.cisco.com/security_services/ciag/initiatives/research/projectsummary.html

Whacking the BGP Session

- Four Macro Ways you can Whack the BGP Session:
  - Saturate the Receive Path Queues: BGP times out
  - Saturate the link: link protocols time out
  - Drop the TCP session
  - Drop the IGP causing a recursive loop up failure

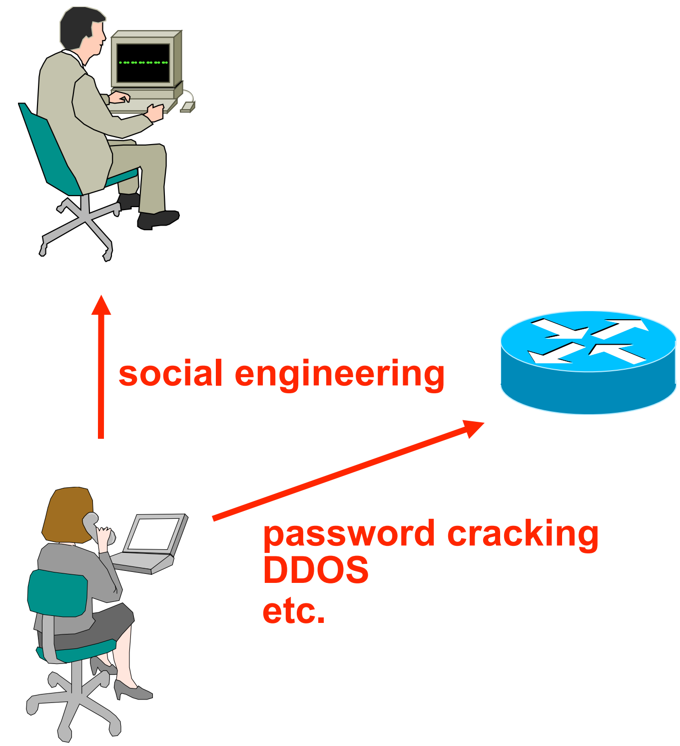
Attacking Routing Devices

- All the normal host attack methods apply to routers
  - Social engineering
  - Password cracking
  - Denial of service
  - etc.
- What an attacker needs:
  - Access to the router
  - (or)
  - Access to the network



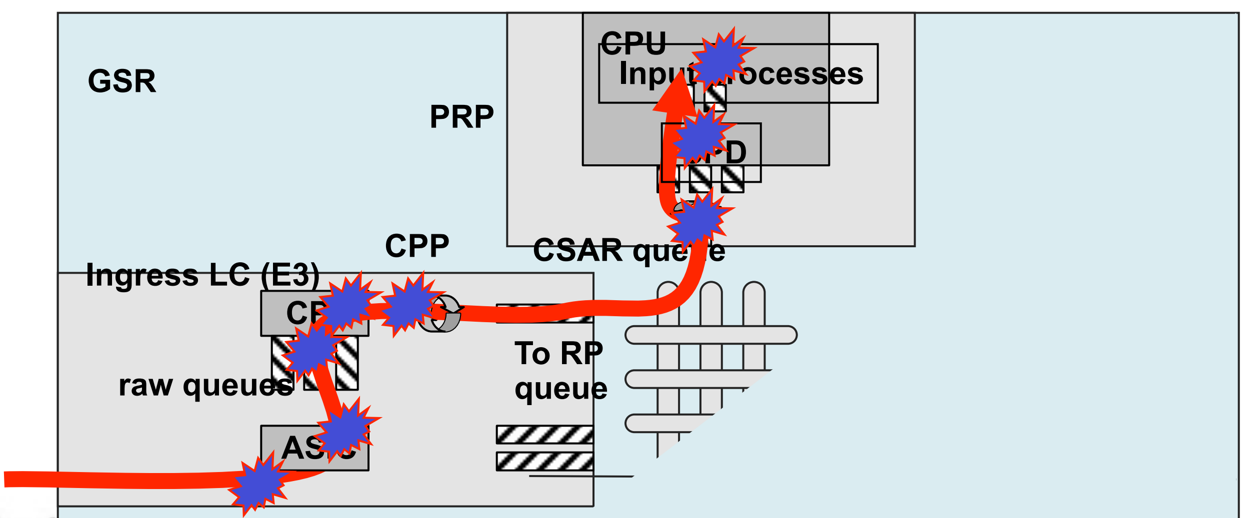
Attacking Routing Devices

- All the normal host attack methods apply to routers
  - Social engineering
  - Password cracking
  - Denial of service
  - etc.
- What an attacker needs:
  - Access to the router
  - (or)
  - Access to the network



Saturate the Receive Path Queues

- Routers usually have various receive path queues that are hit as the packet heads for the TCP Stack.
- Saturation Attacks fill these queues – knocking out valid packets from the queues.
- Consequence: BGP Times out – Dropping the BGP Session



Saturate the Link

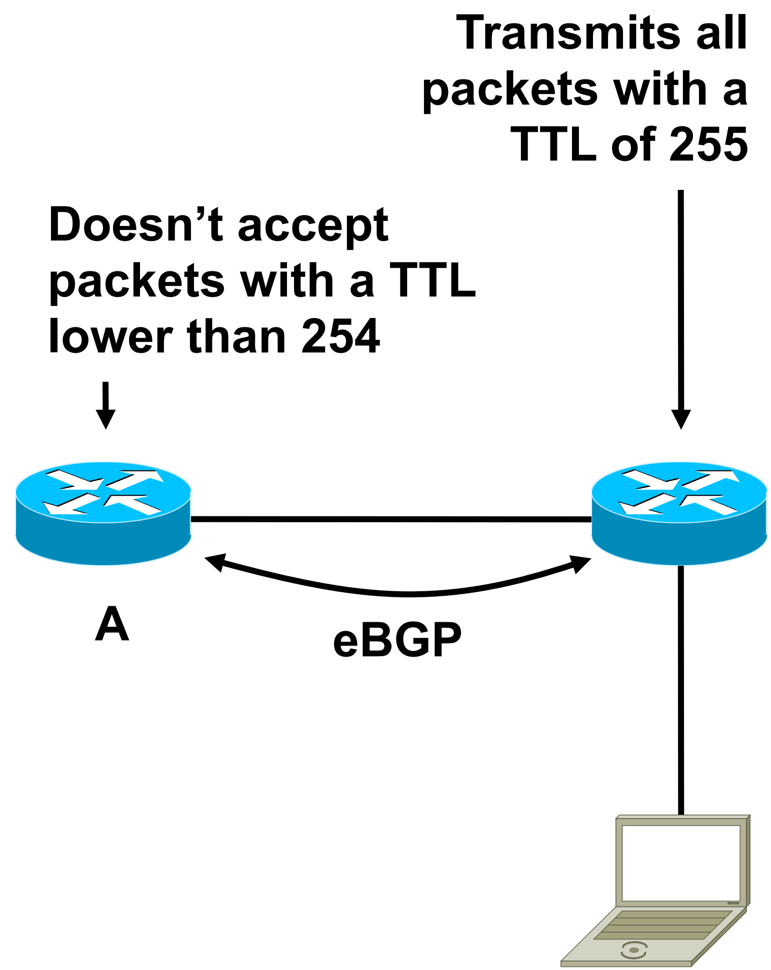
- DOS Attacks Saturating the link will knock out valid control plane packets.
- Link packet over POS, ATM, or Ethernet will drop out – which drop out the link – which drop out the FIB's next hop – which knocks out the BGP Entries
- This is a very effective brute force attack.

Drop the TCP Session

- Dropping the TCP Session was thought to require a breath of packets.
- TCP Session can be dropped with a RST or a SYN (per RFC).
- Successful L4 Spoof is required
  - Match source address
  - Match source port
  - Match destination address (obvious)
  - Match destination port
  - Match Sequence Number (now just get inside the window)

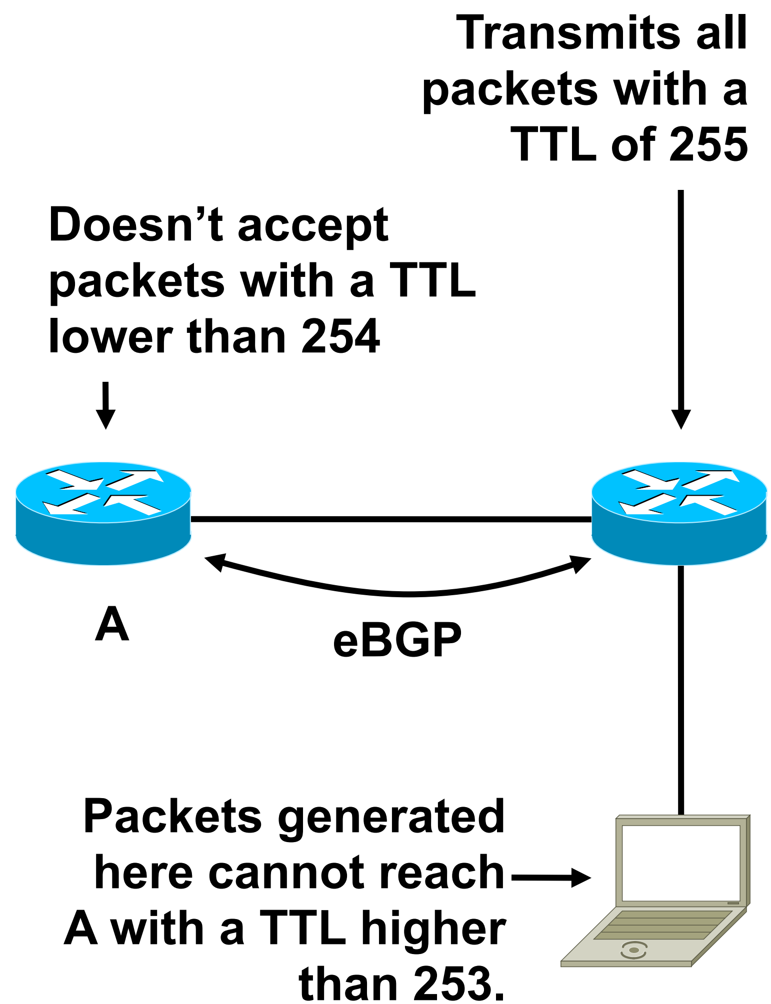
Generalized TTL Security Mechanism

- GTSH is a hack which protects the BGP peers from multihop attacks.
- Routers are configured to transmit their packets with a TTL of 255, and to reject all packets with a TTL lower than 254 or 253.
- A device which isn't connected between the routers cannot generate packets which will be accepted by either one of them.



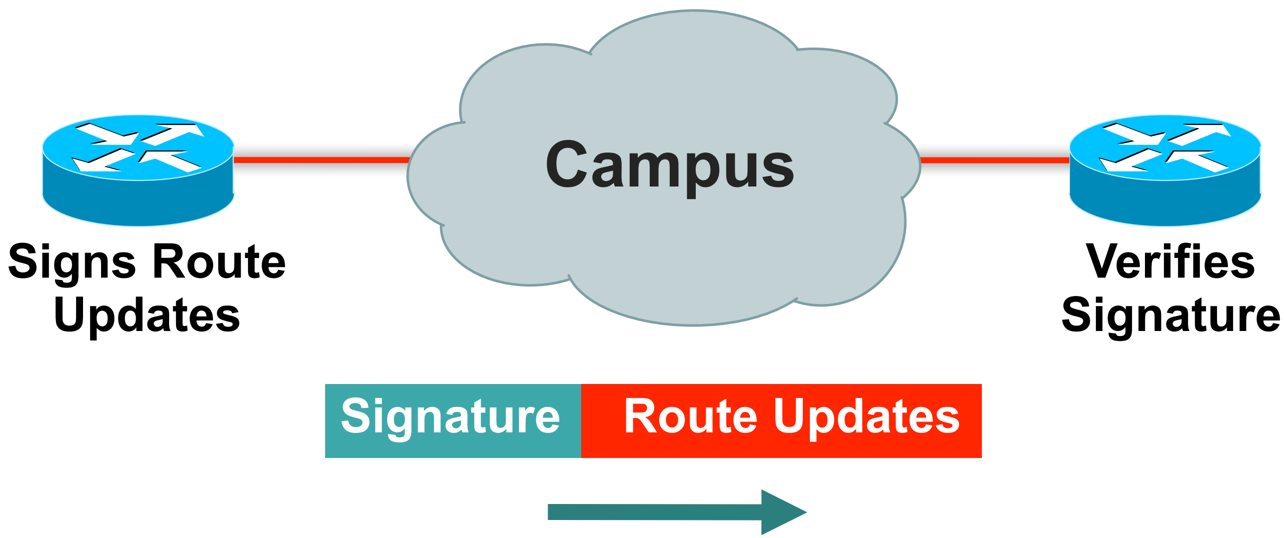
Generalized TTL Security Mechanism

- GTSH is a hack which protects the BGP peers from multihop attacks.
- Routers are configured to transmit their packets with a TTL of 255, and to reject all packets with a TTL lower than 254 or 253.
- A device which isn't connected between the routers cannot generate packets which will be accepted by either one of them.



Secure Routing
Route Authentication

Configure Routing Authentication



Certifies Authenticity of Neighbor and Integrity of Route Updates

Peer Authentication

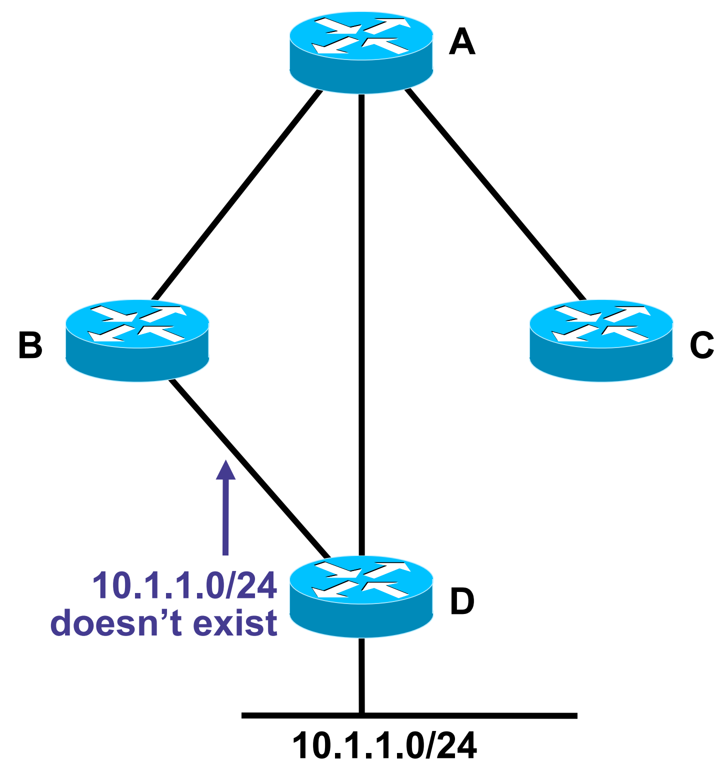
- MD5 Peer authentication can protect against:
  - Malformed packets tearing down a peering session
  - Unauthorized devices transmitting routing information
- MD5 Peer authentication cannot protect against:
  - Reset routing protocol sessions due to denial of service attacks
  - Incorrect routing information being injected by a valid device which has been compromised

Drop the IGP

- Miscreant Success Principle - If you cannot take out the target, move the attack to a coupled dependency of the target.
- BGP's coupled dependency is the IGP it requires for recursive look-up.
- EIGRP and OSPF are both open to external attacks.

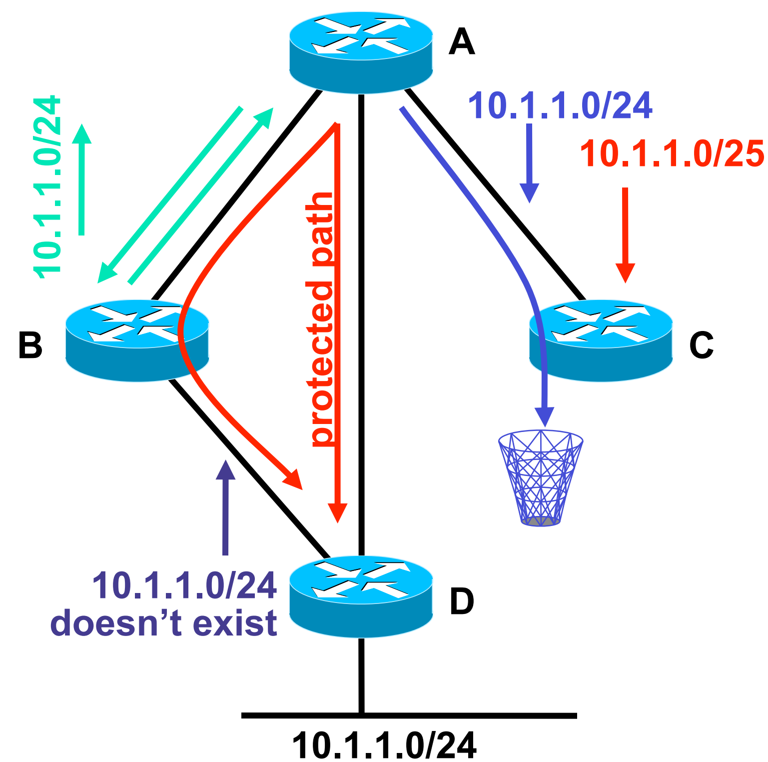
Attacking Routing Data

- How could you attack routing data?
- Modification
  - Direct traffic along an unprotected path
  - Direct traffic into a black hole
  - Create a routing loop
- Overclaiming
  - Injecting nonexistant destinations
  - A longer prefix!
- Underclaiming
  - Removing destinations

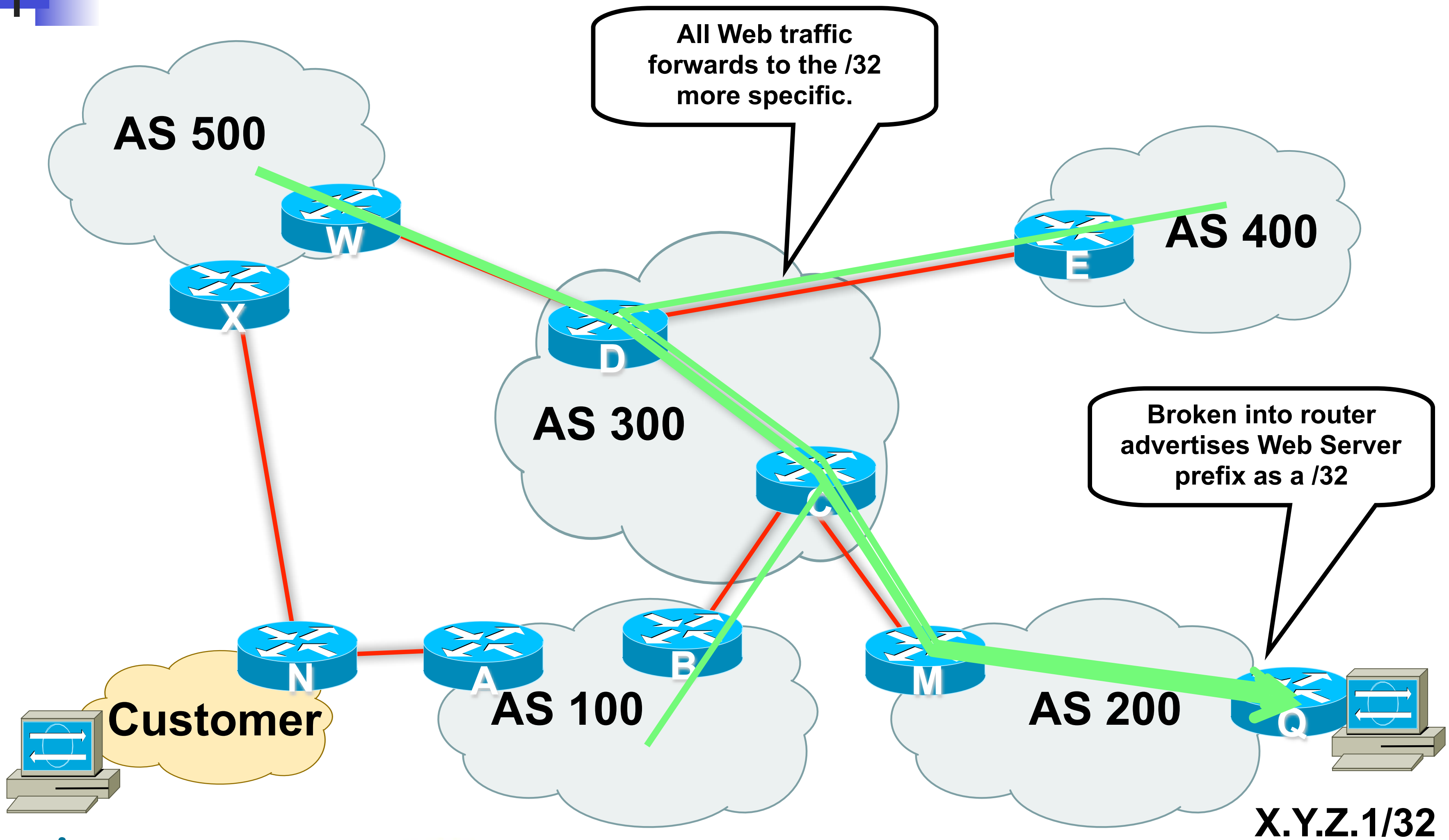


Attacking Routing Data

- How could you attack routing data?
- Modification
  - Direct traffic along an unprotected path
  - Direct traffic into a black hole
  - Create a routing loop
- Overclaiming
  - Injecting nonexistant destinations
  - A longer prefix!
- Underclaiming
  - Removing destinations



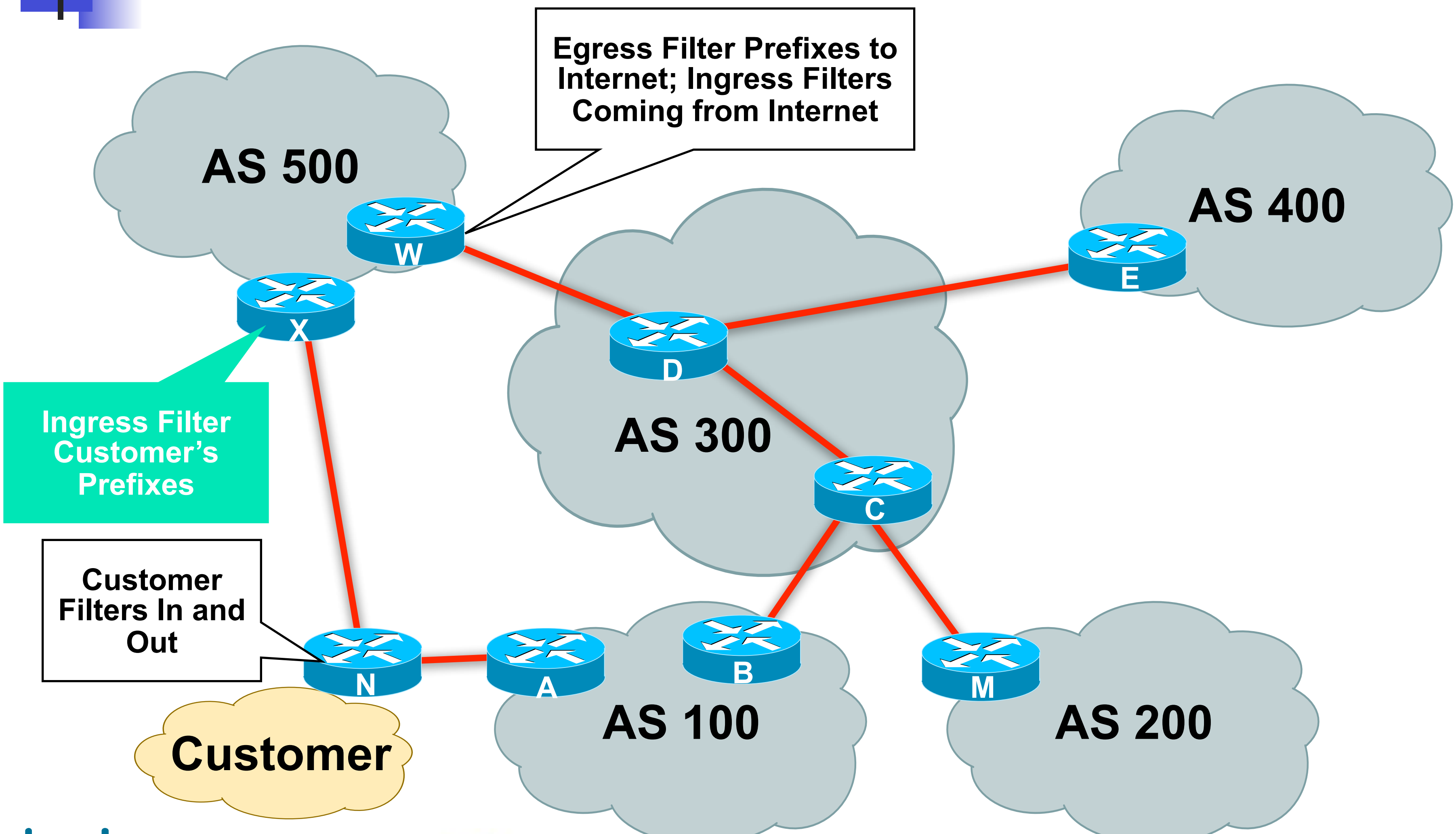
What is a prefix hijack?



Malicious Route Injection
What can ISPs Do?

- Customer Ingress Prefix Filtering!
- ISPs should only accept customer prefixes which have been assigned or allocated to their downstream customers.
- For example
  - Downstream customer has 220.50.0.0/20 block.
  - Customer should only announce this to peers.
  - Upstream peers should only accept this prefix.

Where to Prefix Filter?



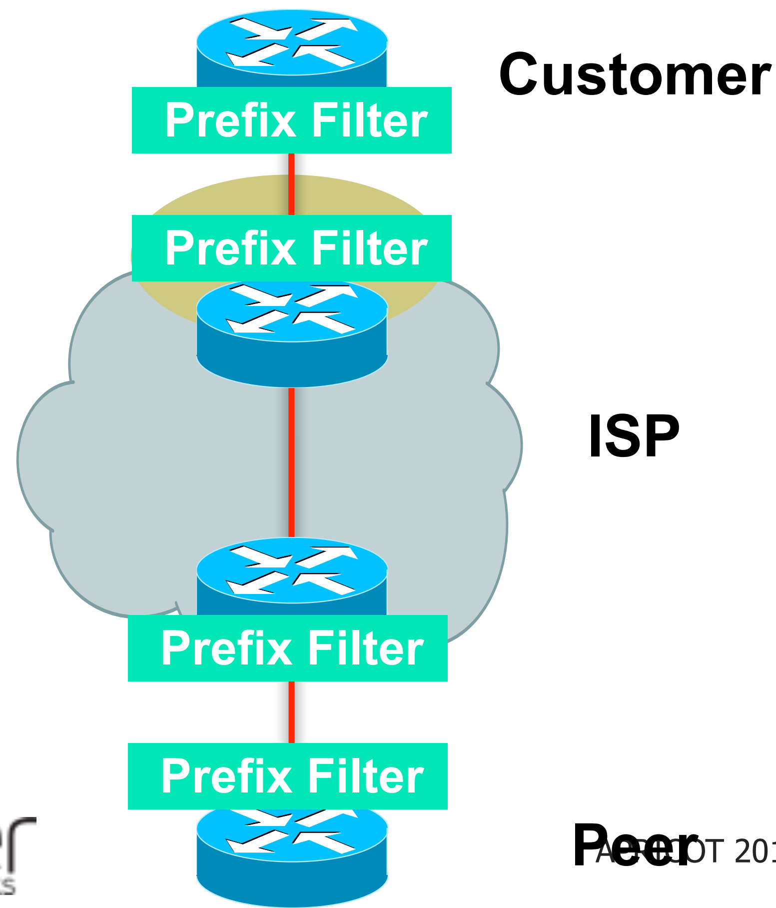
Bogons and Special Use Addresses

- IANA has reserved several blocks of IPv4 that have yet to be allocated to a RIR:
  - http://www.iana.org/assignments/ipv4-address-space
- These blocks of IPv4 addresses should never be advertised into the global internet route table
- Filters should be applied on the AS border for all inbound and outbound advertisements
- Special Use Addresses (SUA) are reserved for special use :-)
  - Defined in RFC3330
  - Examples: 127.0.0.1, 192.0.2.0/24

Prefix Filters: Application

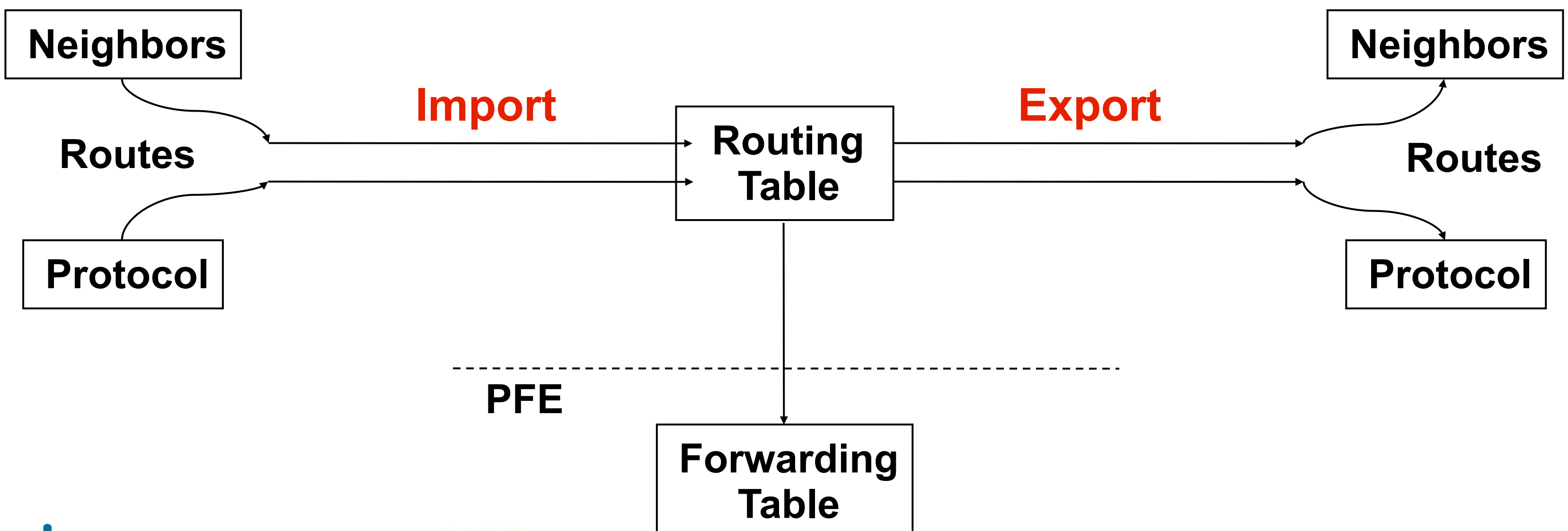
Apply Prefix Filters to All eBGP Neighbors

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



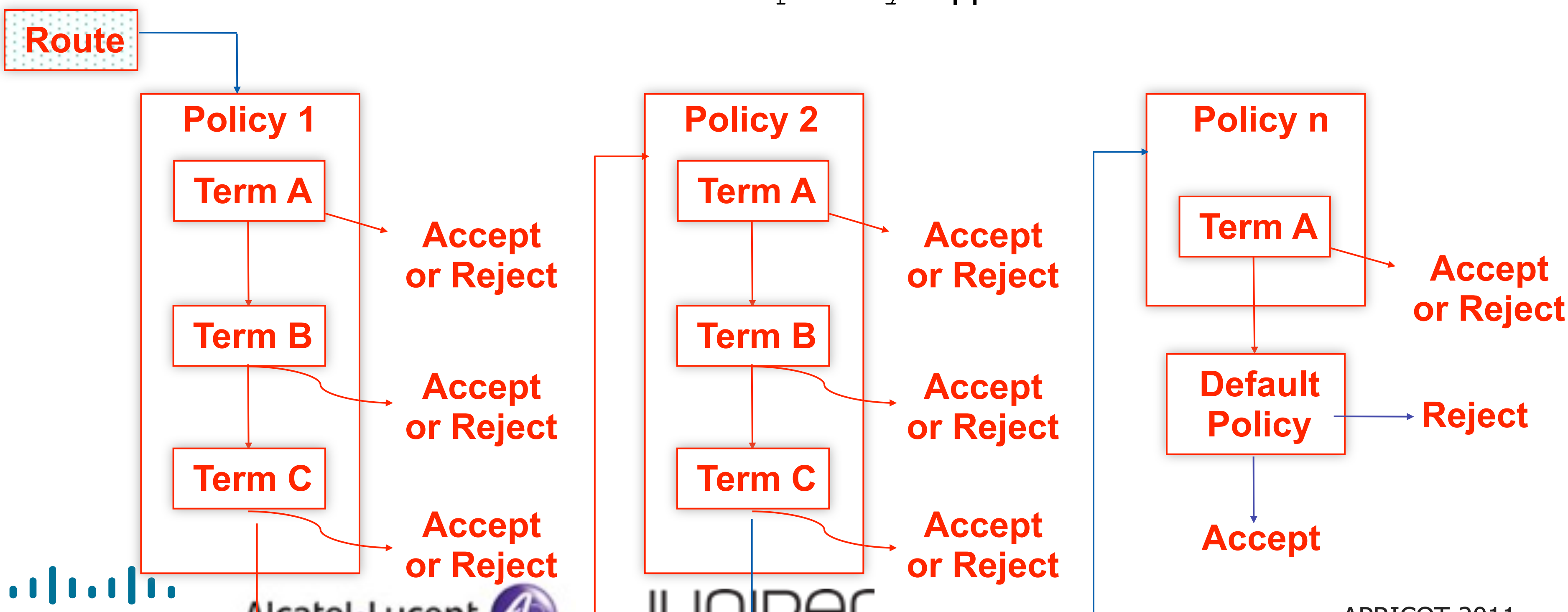
Import and Export Policies

- Perform policy filtering with respect to the software routing table



Routing Policy Flow

- Policies can be chained together
- Evaluation normally proceeds left to right until a terminating action is reached
  - Terminating actions are accept or reject
- Individual policies can contain a collection of terms
  - Flow control actions such as `next-policy` supported



Generic Policy Syntax

Basic policy syntax:

```
policy-options {
    policy-statement policy-name {
        term term-name {
            from {
                match-conditions;
            }
            then {
                action;
            }
        }
    }
}
```

A policy can have multiple terms

Match Conditions

- Policies typically contain some form of match criterion
- Possibilities include:
  - Neighbor address
  - Protocol (source of information)
    - BGP, direct, DVMRP, IS-IS, local, MPLS, OSPF, PIM, RIP, static, aggregate
  - Routing protocol information
    - OSPF area ID
    - IS-IS level number
    - BGP attributes
  - Regular expression-based matches for AS path and communities

Match Actions

- The action associated with a given term/policy is performed for matching routes:
  - Terminating actions
    - Accept route
    - Reject (or suppress) route
  - Flow control actions
    - Skip to next policy
    - Skip to next term
  - Modify attributes actions
    - Metric
    - Preference
    - Color
    - Next-hop address

Default Policies

- Every protocol has a default policy
  - The default policy is applied implicitly at the end of the policy chain; can be overridden with `default-action` statement
- IS-IS and OSPF
  - Import: Accept all routes learned from that protocol
    - Technically, accept all LSPs/LSAs flooded by that protocol
  - Export: Reject everything
    - LSP/LSA flooding announces (IS-IS/OSPF) learned and local routes
- RIP
  - Import all learned RIP routes, export nothing
    - RIP requires export policy to announce RIP (or other) routes
- BGP
  - Import all routes learned from BGP neighbors
  - Export all active routes learned from BGP neighbors to all BGP neighbors
    - EBGP-learned routes are exported to all BGP peers
    - IBGP-learned routes are exported to all EBGP peers (assumes logical IBGP

A Policy Example

- Write a policy statement at the `[edit policy-options]` hierarchy:

```
[edit policy-options]
user@host# show policy-statement advertise-ospf
term pick-ospf {
    from protocol ospf;
    then accept;
}
```

- Apply the policy to one or more routing protocol in the `import`, `export`, or both directions:

```
[edit protocols ospf]
user@host# set export advertise-ospf
```

Another Policy Example

Specifying multiple conditions in a `from` statement means that all criteria must match before the action is taken

```
[edit]
user@host# show policy-options
policy-statement isis-level2 {
    term find-level2-routes {
        from {
            protocol isis;
            level 2;
        }
        then accept;
    }
}
```

Logical AND Function

Applying Policy

- You must apply policies before they can take effect
- Link-state protocols (IS-IS and OSPF) have only export filtering points
- BGP and RIP support both import and export policies

```
[edit protocols]
user@host# show
bgp {
    import bgp-import;
    export bgp-export;
}
ospf {
    export ospf-export;
}
```

Apply Routing Policy to BGP

- BGP has three filtering points per direction:
  - Global
  - Groups of neighbors
  - Individual neighbors
- Only the most specific policies are applied to a particular peer
  - Neighbor policy overrides group and global policies
  - Group policy overrides global policy

BGP Policy Application Example

```
[edit protocols]
user@host# show
bgp {
    export local-customers;
    group meganet-inc {
        type external;
        import [ martian-filter long-prefix-filter as-47-filter ];
        peer-as 47;
        neighbor 1.2.2.4;
        neighbor 1.2.2.5;
    }
    group problem-child {
        type external;
        import [ as-47-filter long-prefix-filter martian-filter ];
        export kill-private-addresses;
        peer-as 54;
        neighbor 1.2.2.6;
        neighbor 1.2.2.7;
        neighbor 1.2.2.8 {
```

Route Filters

- Use route filters to match an individual route (or groups of routes)
  - You can specify multiple route filters within a single term
  - General syntax in the form of:

`route-filter` *prefix/prefix-length* *match-type* *actions*;

- Route filter evaluation has special rules according to the match type
  - Match types specify different sets of routes:
    - `exact`
    - `orlonger`
    - `longer`
    - `upto`
    - `through`
    - `prefix-length-range`
  - Policy `test` function is useful for route-filter debugging

Route Filter Match Types (2 of 5)

- `orlonger`
  - Match the specified prefix and mask exactly
  - Also match any routes that start with the same prefix and have longer masks

```
from route-filter 192.168/16 orlonger;
```

	200.1.1.0/24		192.168.0.0/24
	192.0.0.0/8		192.168.1.0/24
	192.168.0.0/16		192.168.64.0/25
	192.168.1.0/16		192.168.32.0/26
	192.168.0.0/17		192.168.192.0/26
	192.168.128.0/17		192.168.1.1/32

Route Filter Match Types (3 of 5)

- `longer`
  - Do not match the specified prefix and mask exactly
  - Match only the routes that start with the same prefix and have longer masks

```
from route-filter 192.168/16 longer;
```

	200.1.1.0/24		192.168.0.0/24
	192.0.0.0/8		192.168.1.0/24
	192.168.0.0/16		192.168.64.0/25
	192.168.1.0/16		192.168.32.0/26
	192.168.0.0/17		192.168.192.0/26
	192.168.128.0/17		192.168.1.1/32

Route Filter Match Types (4 of 5)

- `upto`
  - Match the specified prefix and mask exactly
  - Also match any routes that start with the same prefix and have a mask no longer than the second value specified

```
from route-filter 192.168/16 upto /24;
```

	200.1.1.0/24		192.168.0.0/24
	192.0.0.0/8		192.168.1.0/24
	192.168.0.0/16		192.168.64.0/25
	192.168.1.0/16		192.168.32.0/26
	192.168.0.0/17		192.168.192.0/26
	192.168.128.0/17		192.168.1.1/32

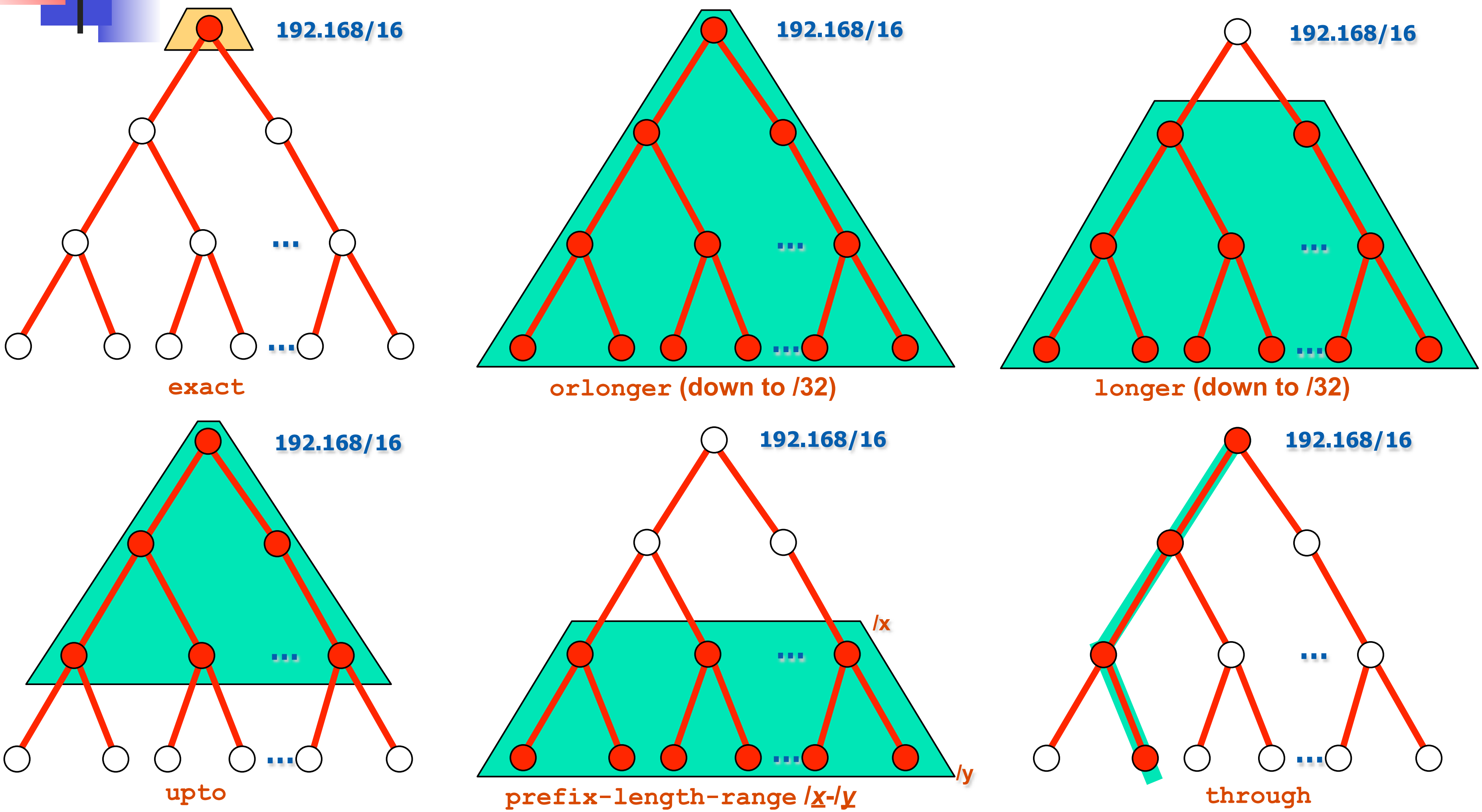
Route Filter Match Types (5 of 5)

- `prefix-length-range`
  - Match only routes that start with the same prefix and have a mask between the two values specified (inclusive match)

```
from route-filter 192.168/16 prefix-length-range /20-/24;
```

	200.1.1.0/24		192.168.0.0/22
	192.0.0.0/8		192.168.1.0/24
	192.168.0.0/16		192.168.64.0/25
	192.168.1.0/16		192.168.32.0/26
	192.168.0.0/17		192.168.192.0/26
	192.168.196.0/20		192.168.1.1/32

Match Types Summary



Route Filter Actions

```
term term-name {
   from {
       route-filter dest-prefix match-type actions;
       route-filter dest-prefix match-type actions;
   }
   then actions;
}
```

Longest-Match Lookup

- Only one route filter in a given term can be considered a match
  - Longest-match lookup is performed on the prefix being evaluated
- If an action is specified to a route filter, it takes effect immediately
  - The global `then` portion of the term is ignored
    - If specific actions are not defined, the `then` portion of the term is

Test Your Knowledge (1 of 2)

Which action is taken when this policy evaluates 10.0.67.43/32?

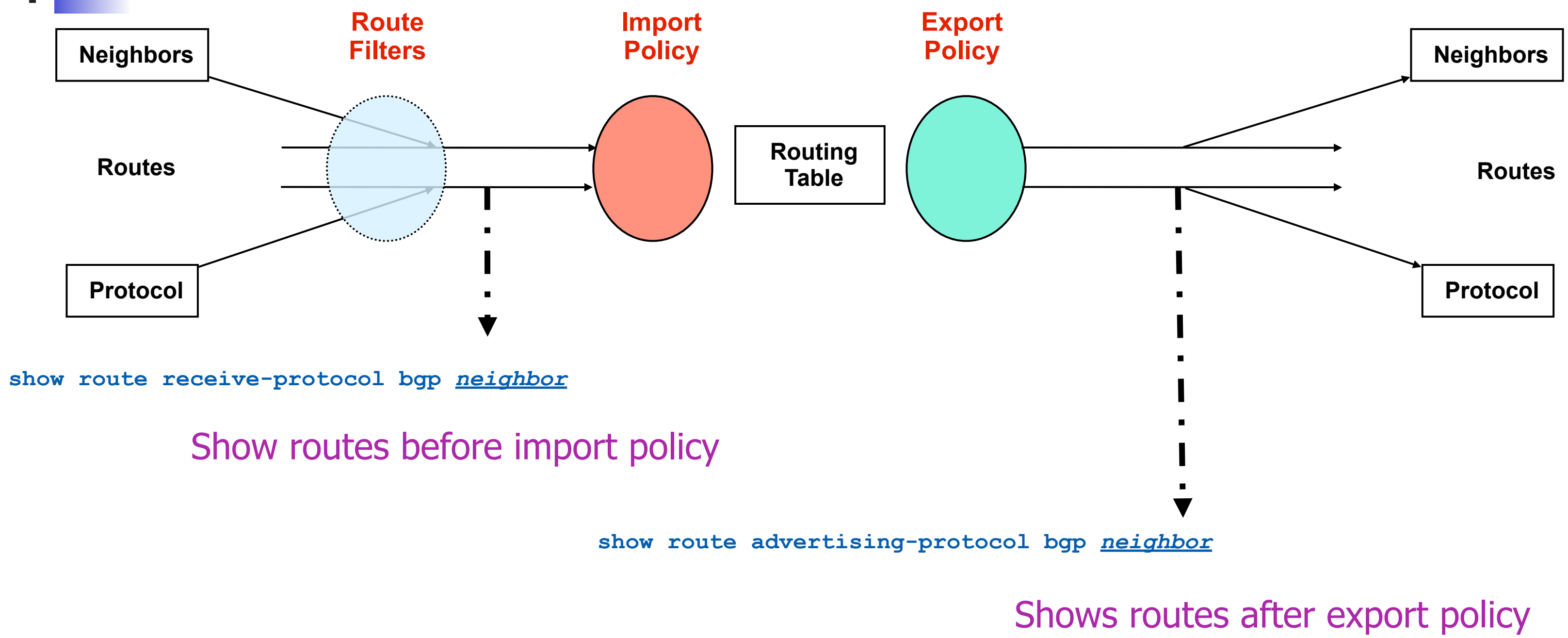
```
[edit policy-options policy-statement pop-quiz]
user@host# show
from {
    route-filter 10.0.0.0/16 orlonger accept;
    route-filter 10.0.67.0/24 orlonger;
    route-filter 10.0.0.0/8 orlonger reject;
}
then {
    metric 10;
    accept;
```

Test Your Knowledge (2 of 2)

Which action is taken when this policy evaluates 10.0.55.2/32?

```
[edit policy-options policy-statement pop-quiz]
user@host# show
from {
    route-filter 10.0.0.0/16 orlonger accept;
    route-filter 10.0.67.0/24 orlonger;
    route-filter 10.0.0.0/8 orlonger reject;
}
then {
    metric 10;
    accept;
```

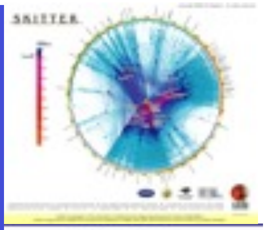
Monitoring Policy Operation

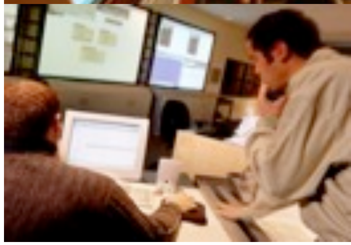


- The `show route receive-protocol` and `show route advertising-protocol` commands:

Lab: Securing routing protocols

Remote Trigger Black Hole [RTBH]

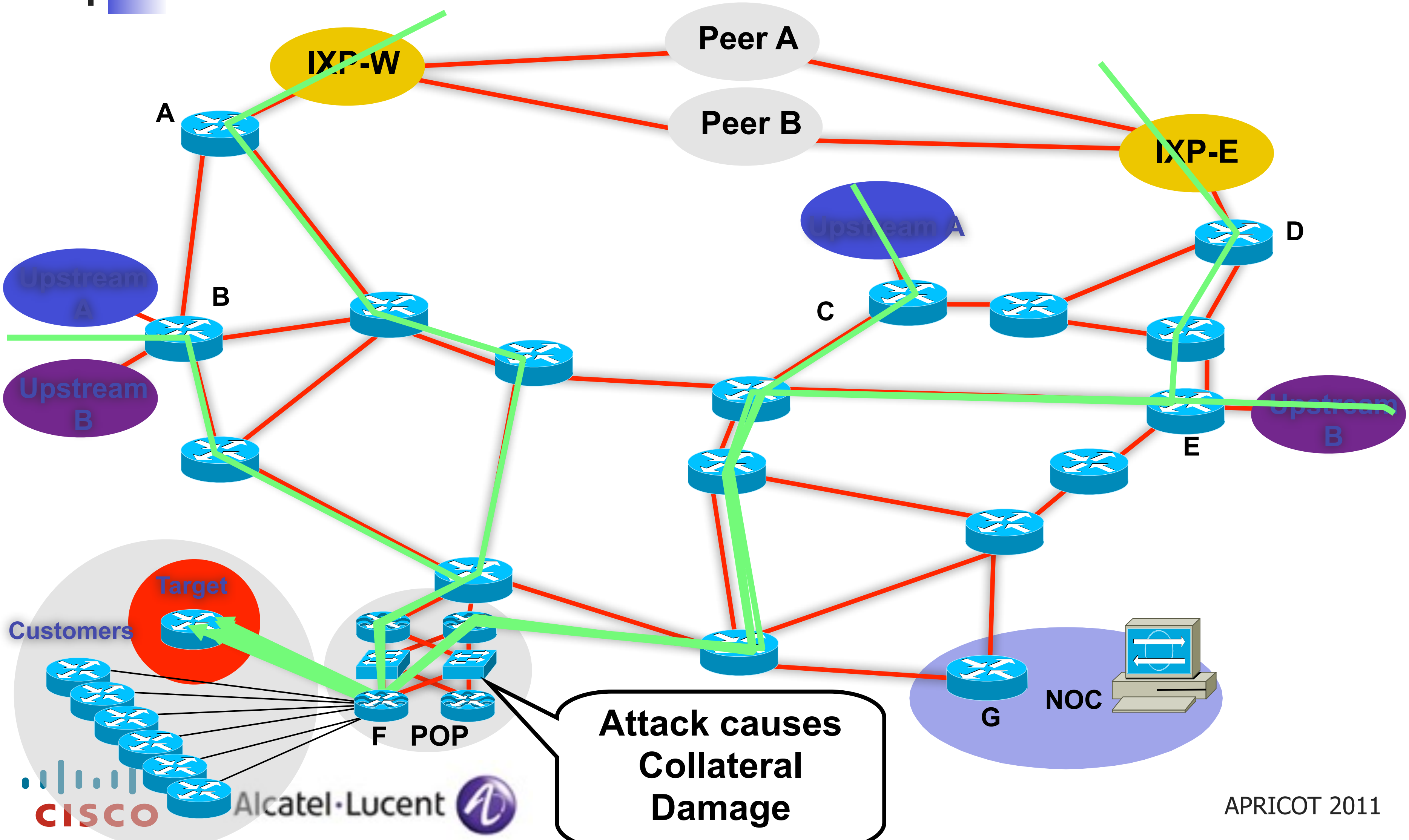




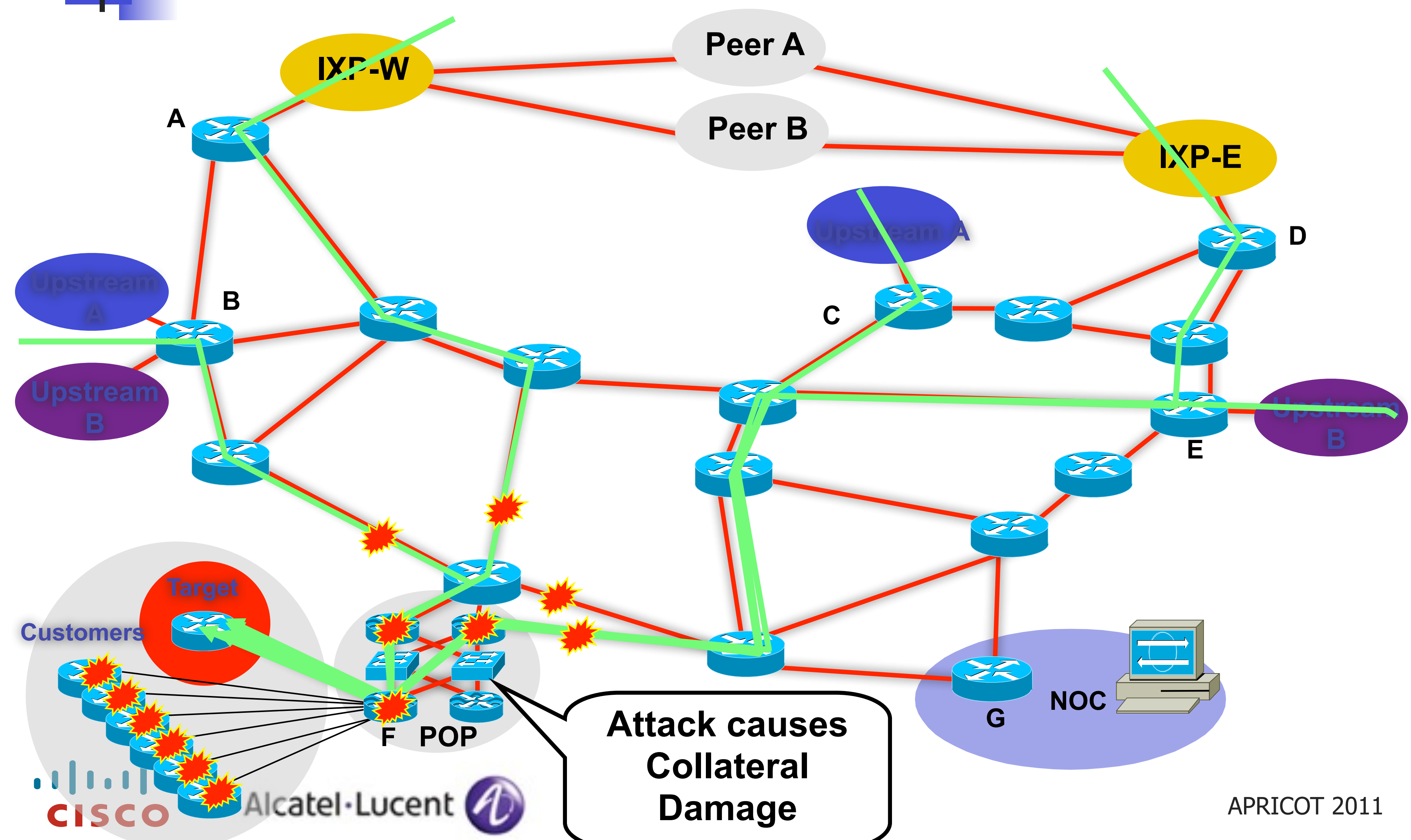
RTBH Filtering

- We use BGP to trigger a network wide response to a range of attack flows.
- A simple static route and BGP will allow an ISP to trigger network wide black holes 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 – Collateral Damage



Customer is DOSed – Before – Collateral Damage



Shunning with uRPF and BGP

- 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

Discard Interface

- JUNOS supports the use of a ‘discard’ interface
  - Can assign filters to get granular filters, counting and sampling logs too

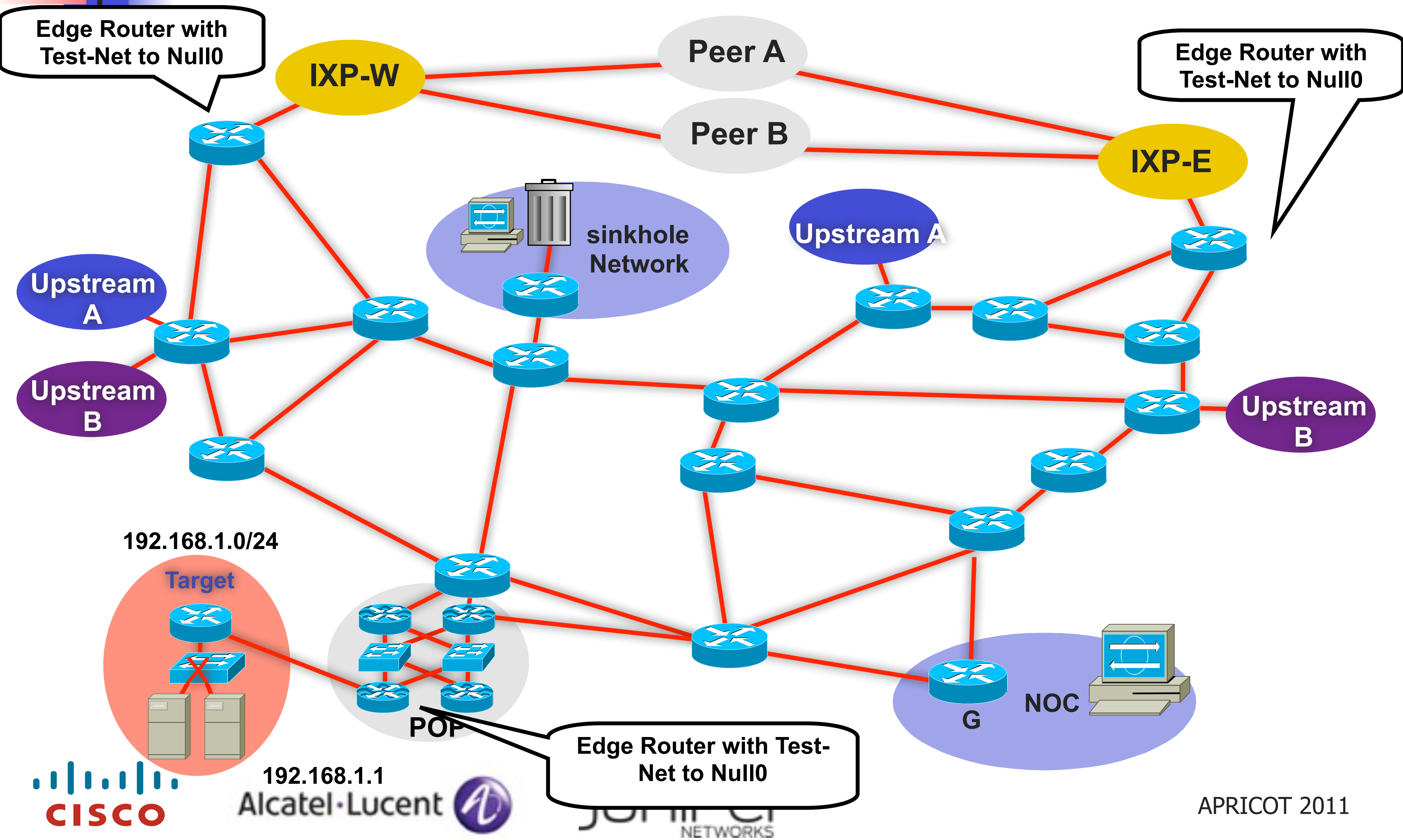
```
Show configuration interfaces dsc
Unit 0 {
   family inet {
        address 192.168.1.1/32 {
                destination 192.168.1.2;
        }
        address 192.168.1.3/32 {
                destination 192.168.1.4;
        }
}
Sh interface terse
Interface       Admin  Link   Proto  Local  remote
Dsc.0           Up     Up     Inet 192.168.1.1 192.168.1.2
```

Step 1- Prepare all the Routers w/ Trigger

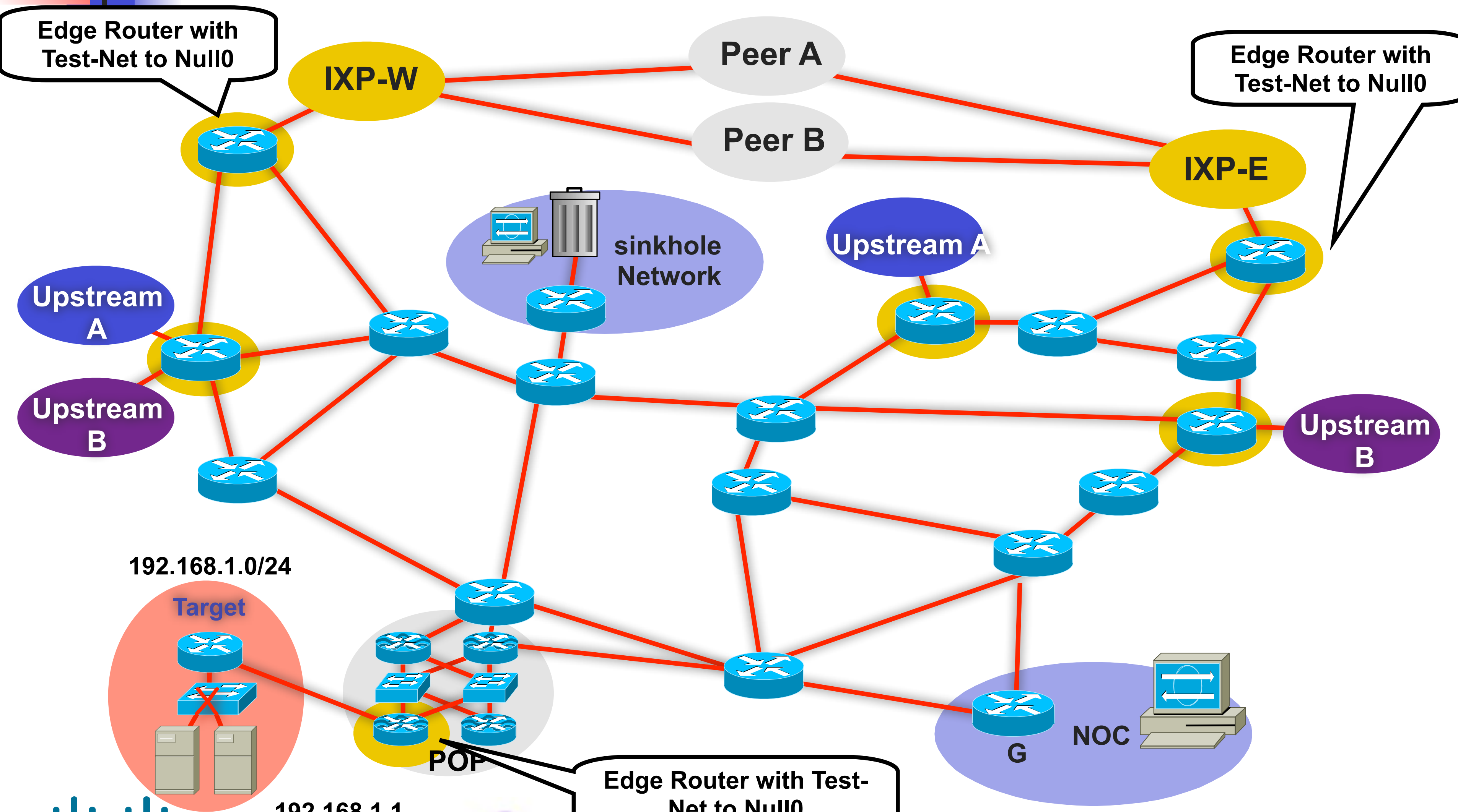
- Select a small block that will not be used for anything other than black hole filtering. Test Net (192.0.2.0/24) is optimal since it should not be on the Net and is not really used.
- Put a static route with Test Net – 192.0.2.0/24 to Null 0 on every edge router on the network.

```
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 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.
- iBGP neighbor - it can (and probably should) be a route-reflector client, no need for full mesh
- Can be a separate router or Arbor's Peakflow DoS anomaly-detection system - if a separate router, something small like a 2600 will do (it doesn't need to receive BGP routes, only send them; use a prefix-list to ensure that it doesn't end up redistributing routes)
- Can be a production router (some ISPs do this, but not the recommended approach)
- Can be a workstation with Zebra/Quagga (interface with Perl scripts and other tools).

Trigger Router's Config

Redistribute Static with a route-map

```
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
```

Match Static Route Tag

Set Next-Hop to the Trigger

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.
  - ip route 192.168.1.1 255.255.255.255 Null0 Tag 66
  - BGP Advertisement goes out to all BGP-speaking routers which peer with the trigger.
- Routers hear the announcement, glue it to the existing static on the route, and changes the next-hop for the BGP advertised route to Null0 – triggering black hole routing.

Activate the Black Hole

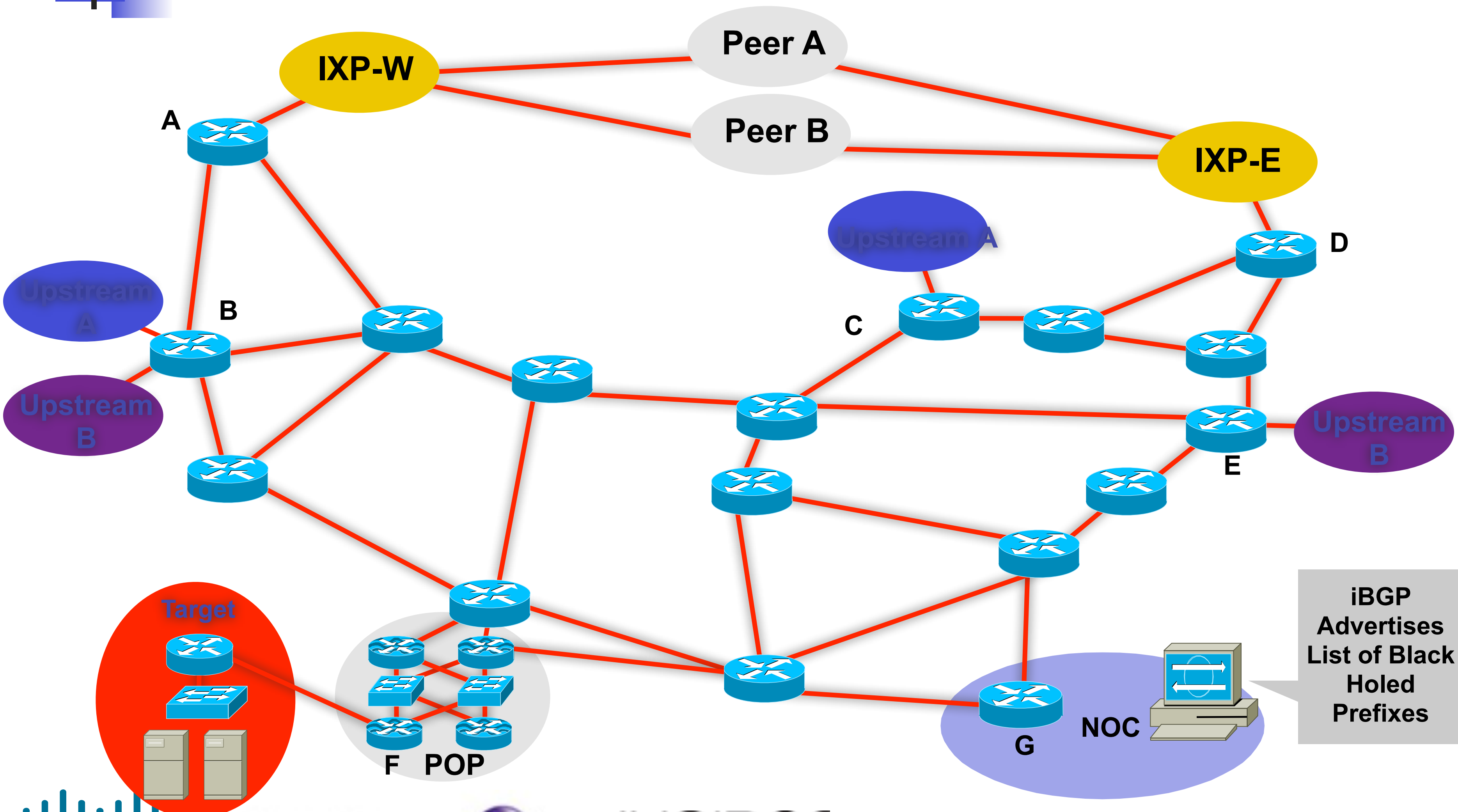
BGP Sent – 192.168.1.1 Next-Hop = 192.0.2.1

Static Route in Edge Router – 192.0.2.1 = Null0

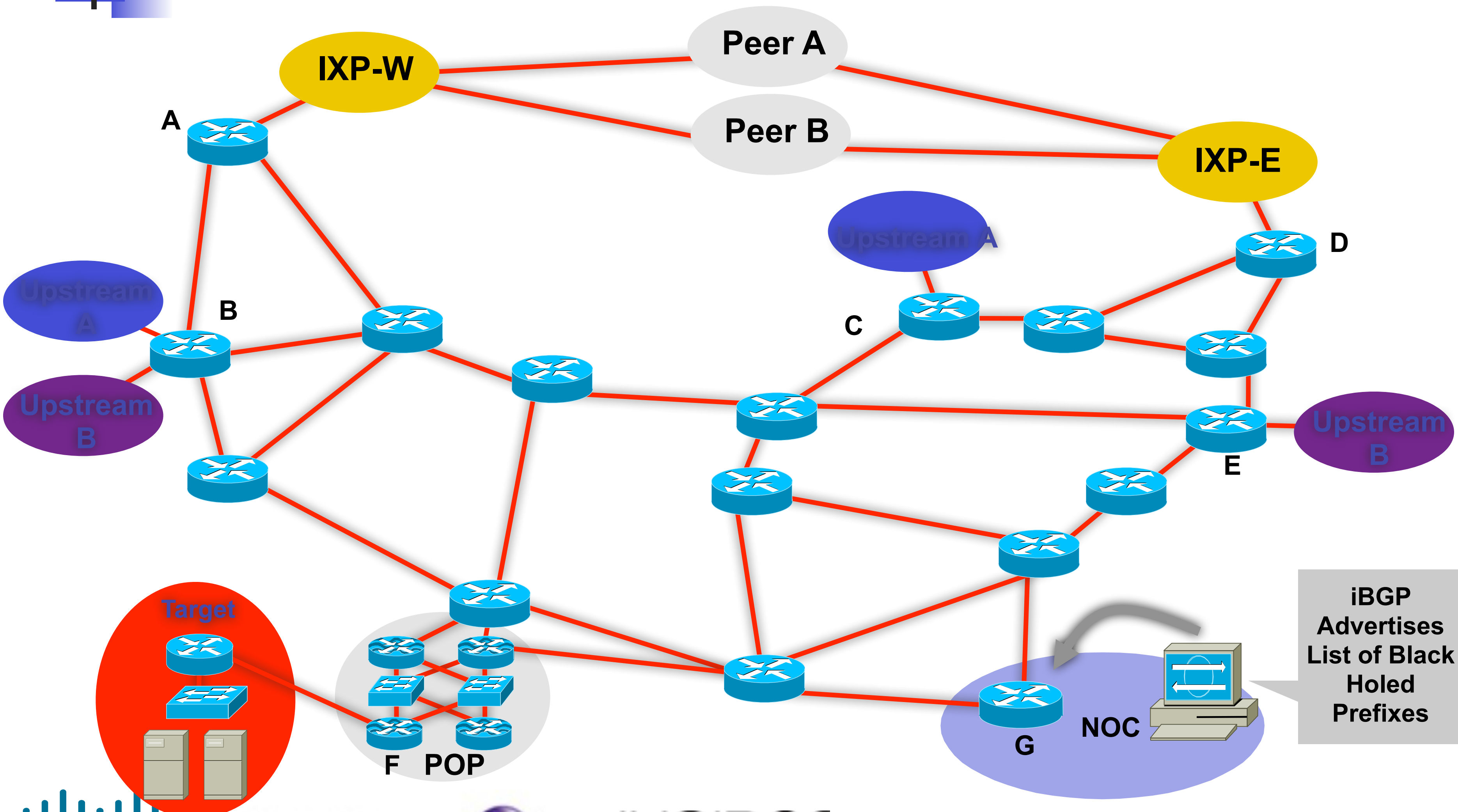
192.168.1.1 = 192.0.2.1 = Null0

Next hop of 192.168.1.1 is now equal to Null0

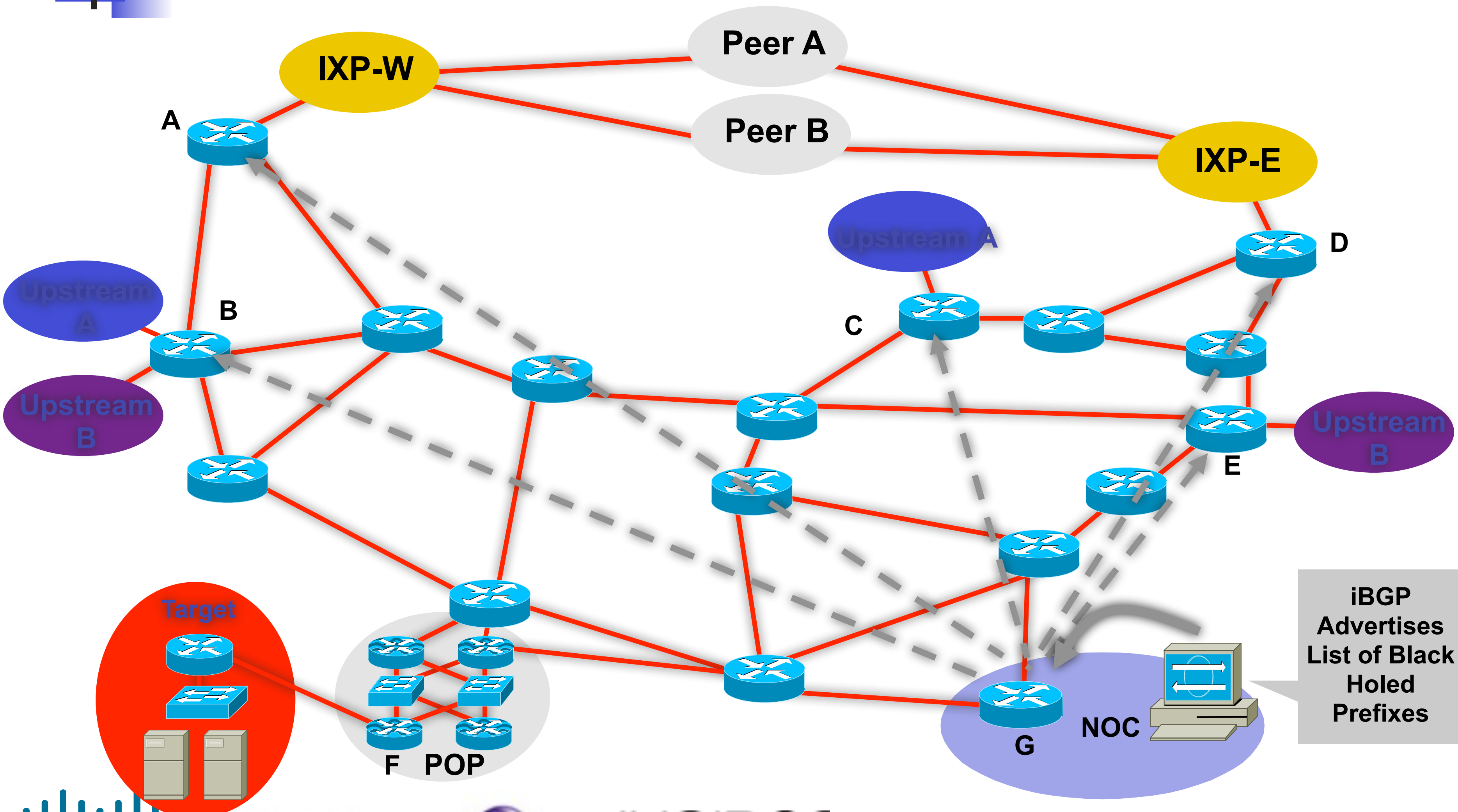
Activate the Black Hole



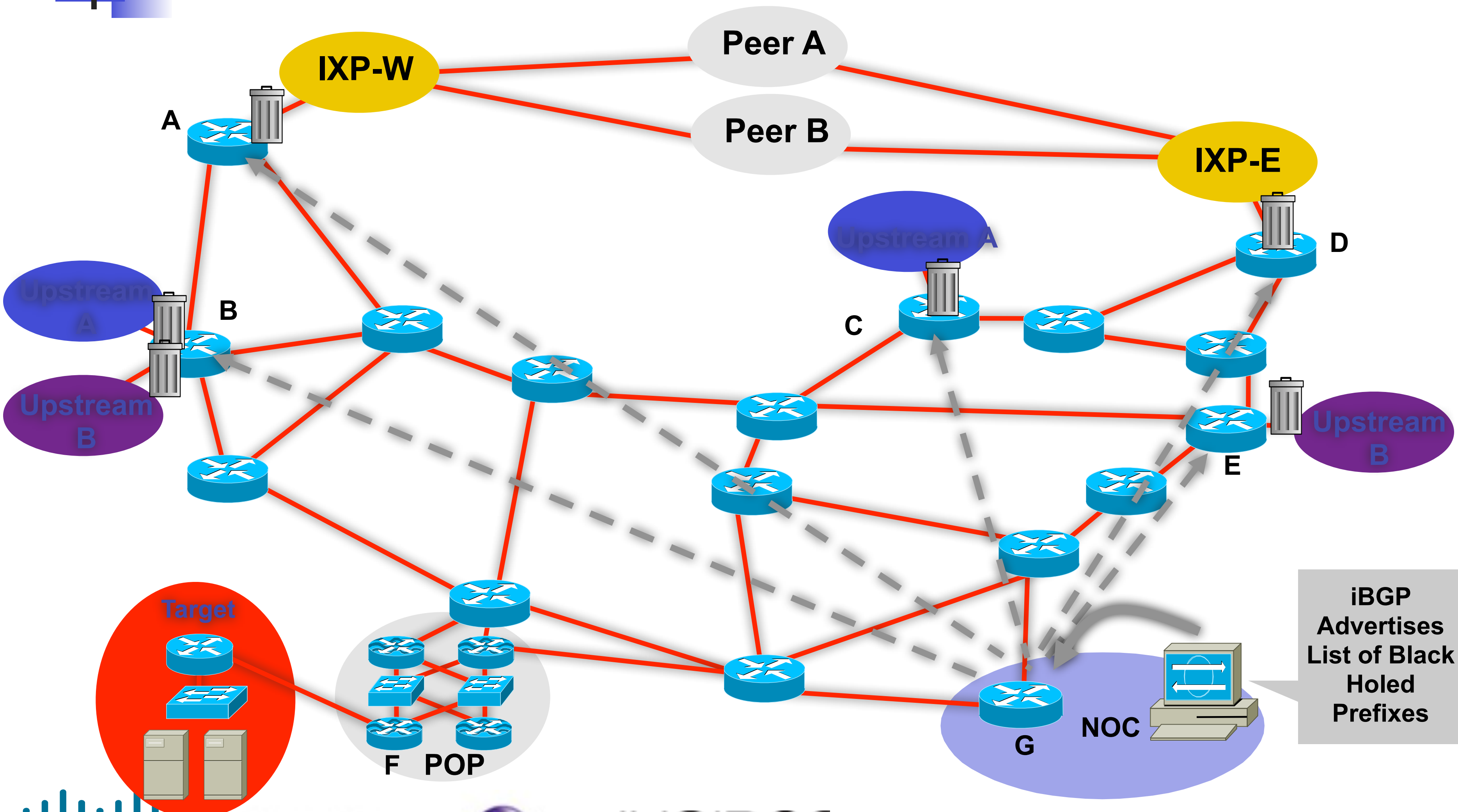
Activate the Black Hole



Activate the Black Hole



Activate the Black Hole

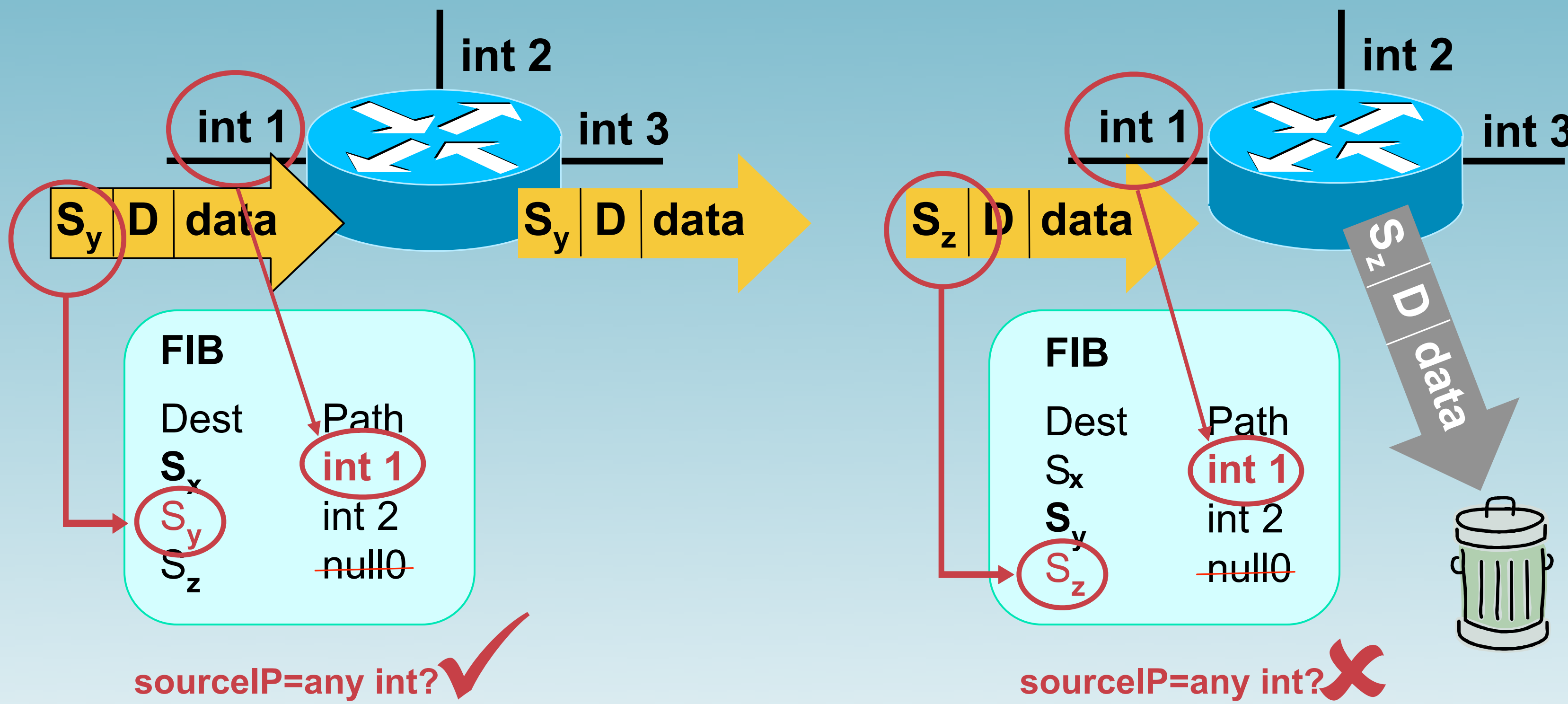


Flipping it Around: Triggered Source Drops

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

uRPF – Loose Mode

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



IP verify unicast source reachable – via any

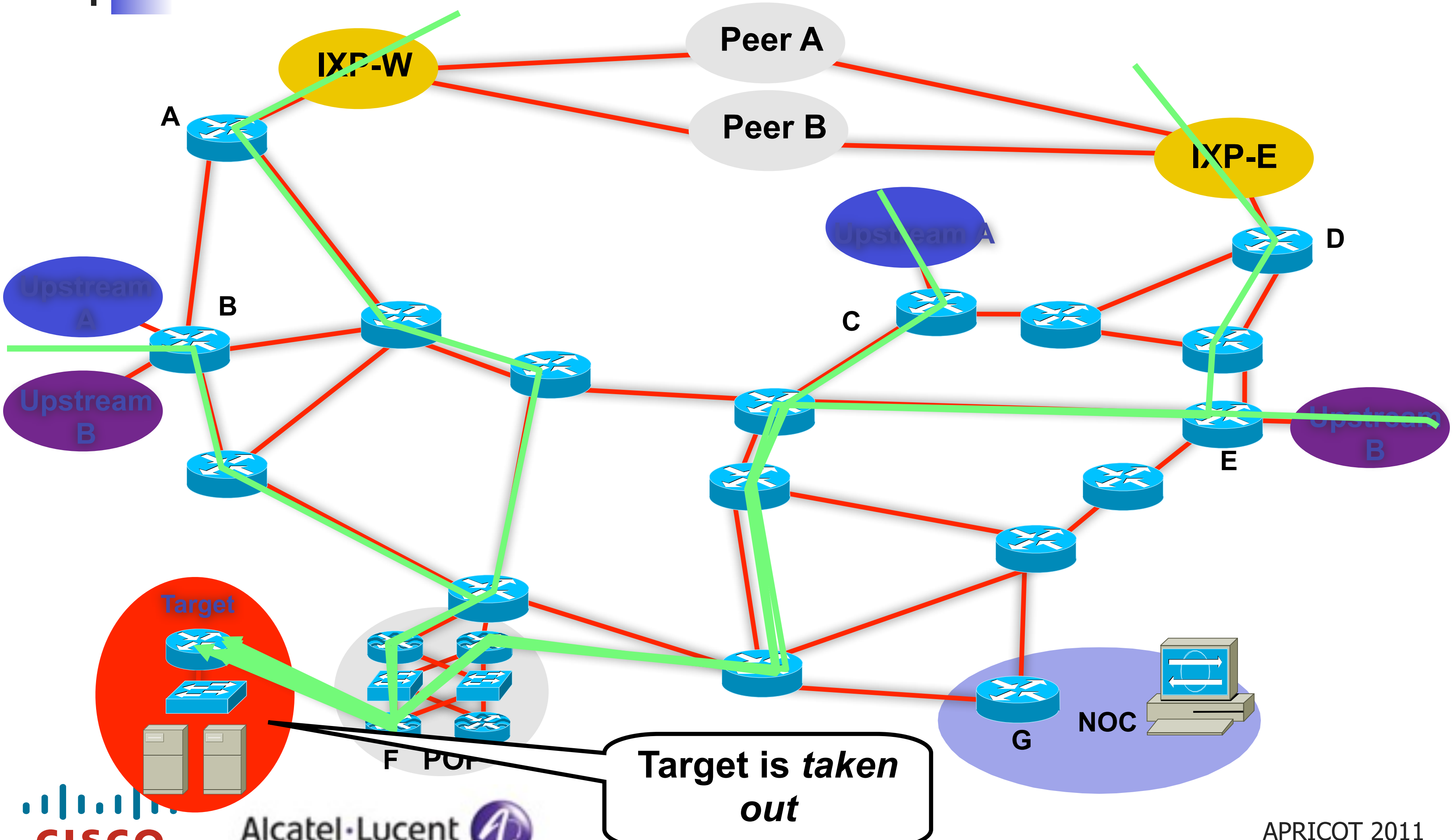
Source Based RTBH Filtering

- 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

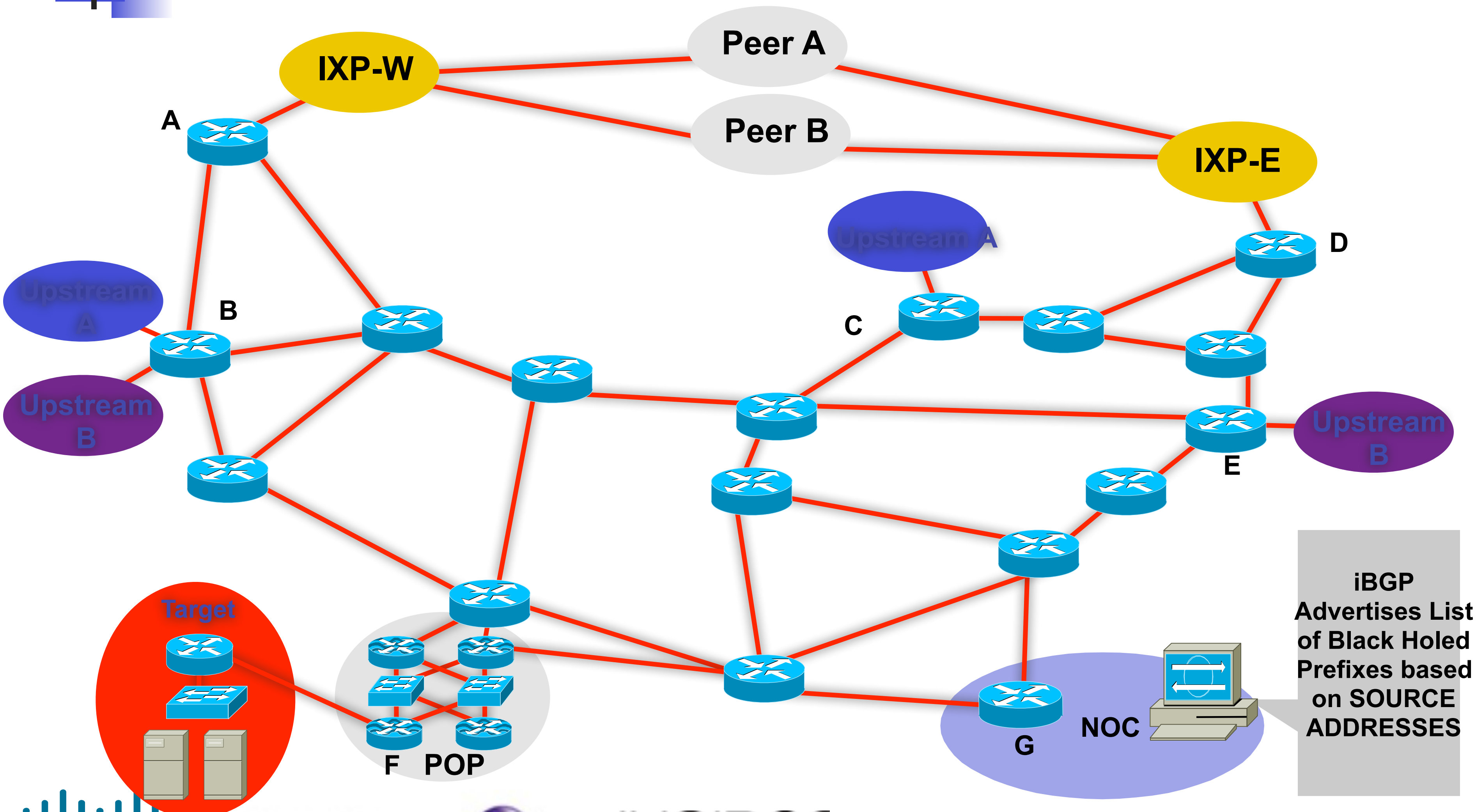
Source Based RTBH Filtering

- 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!

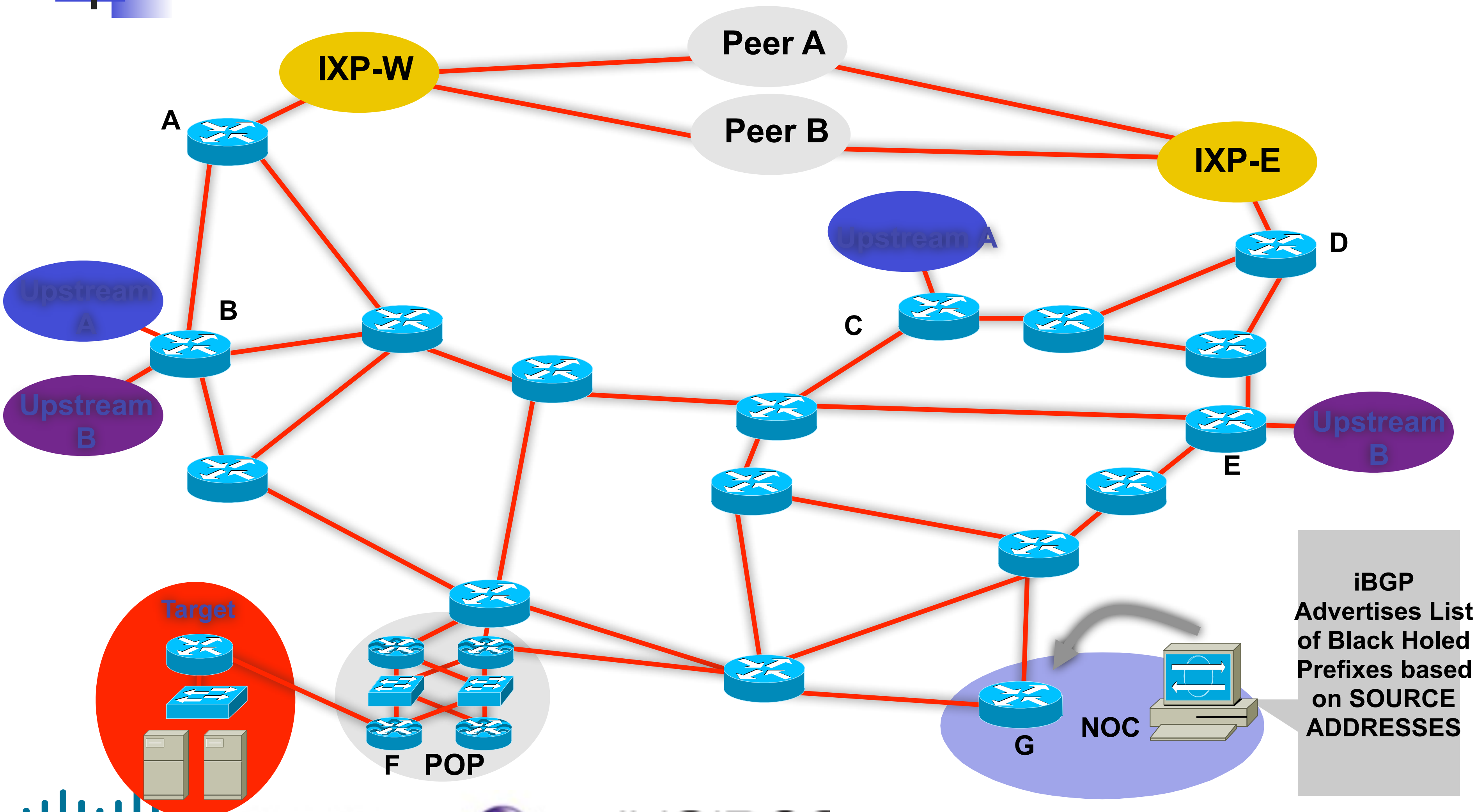
Customer is DOSed - Before



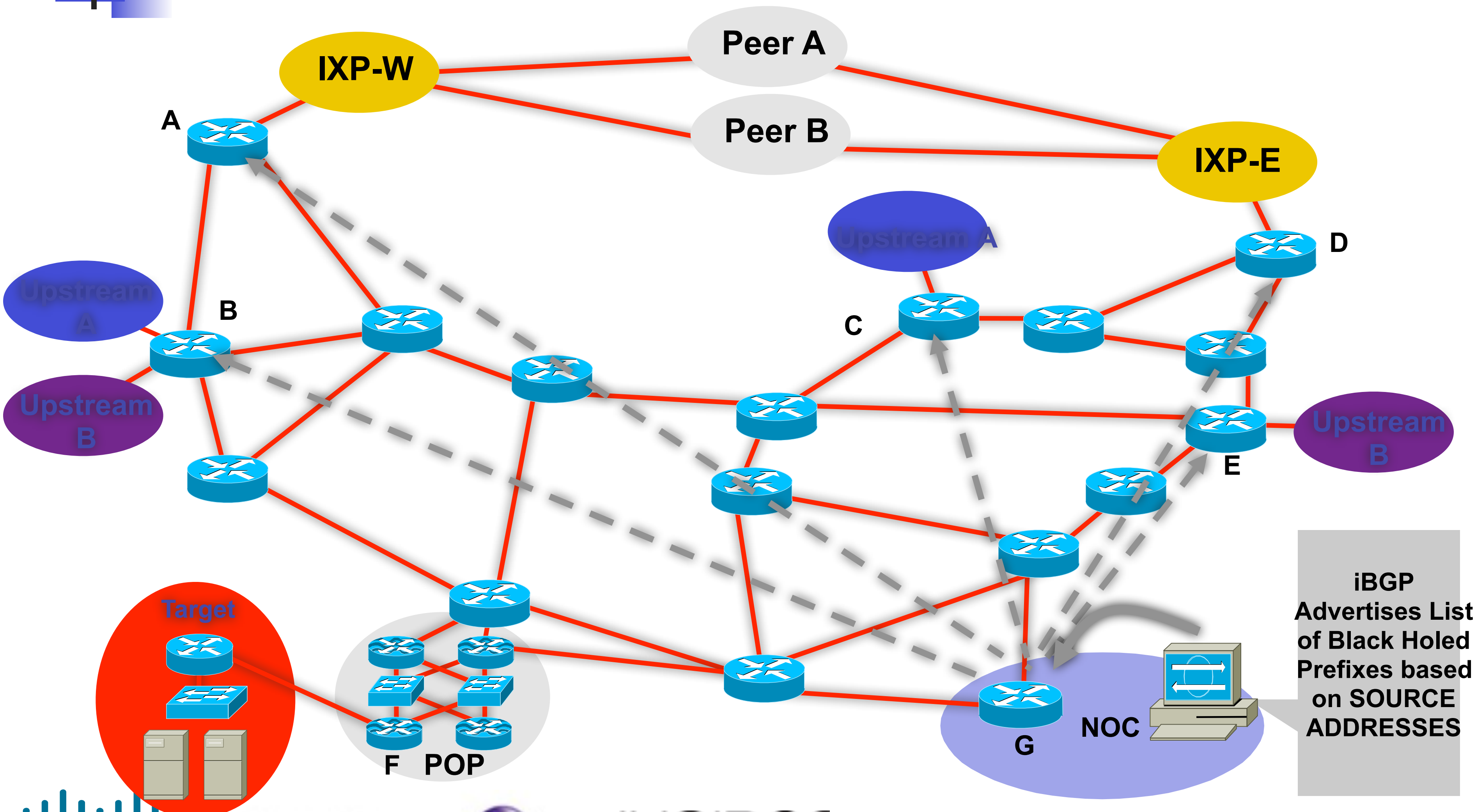
Customer is DOSed – After



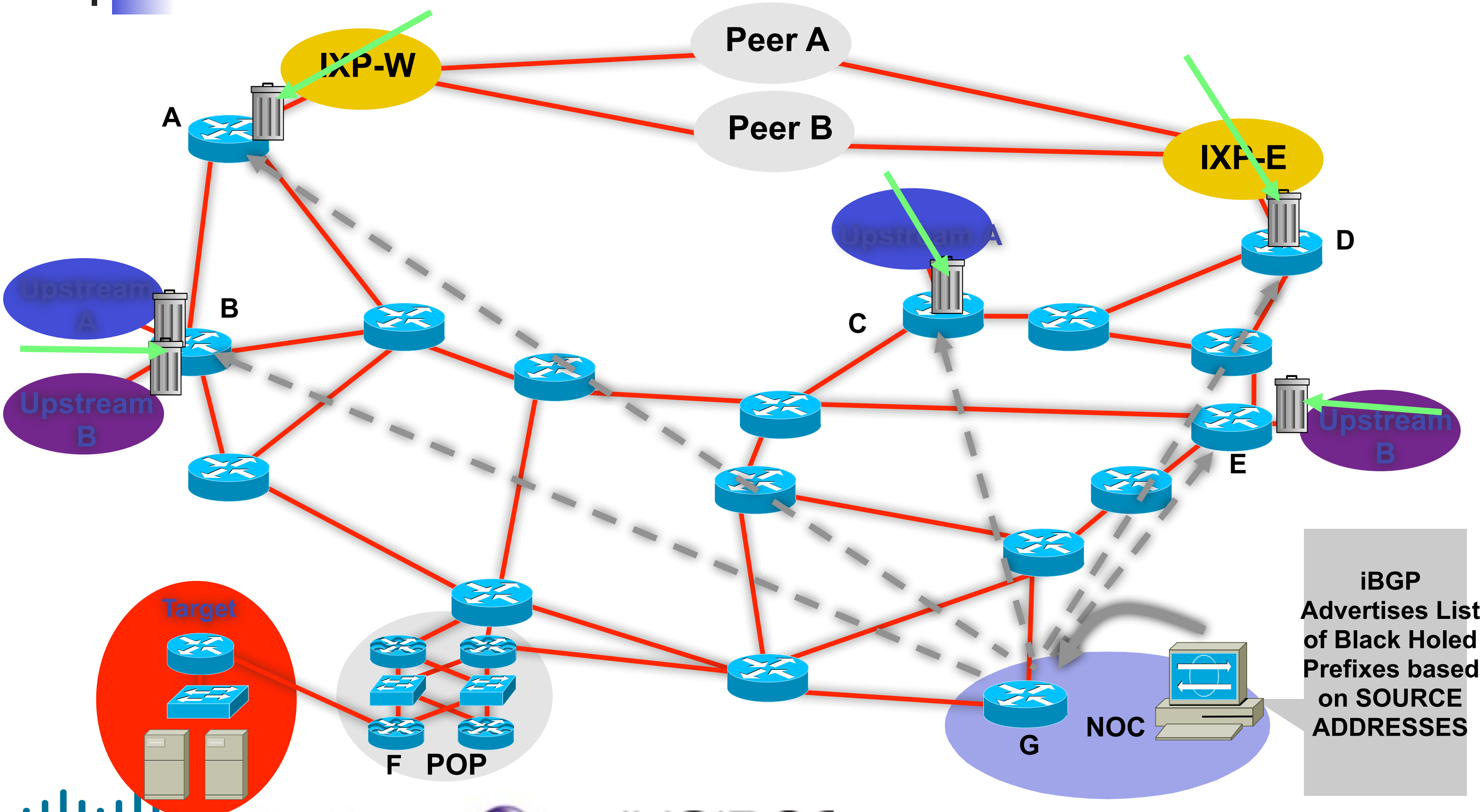
Customer is DOSed – After



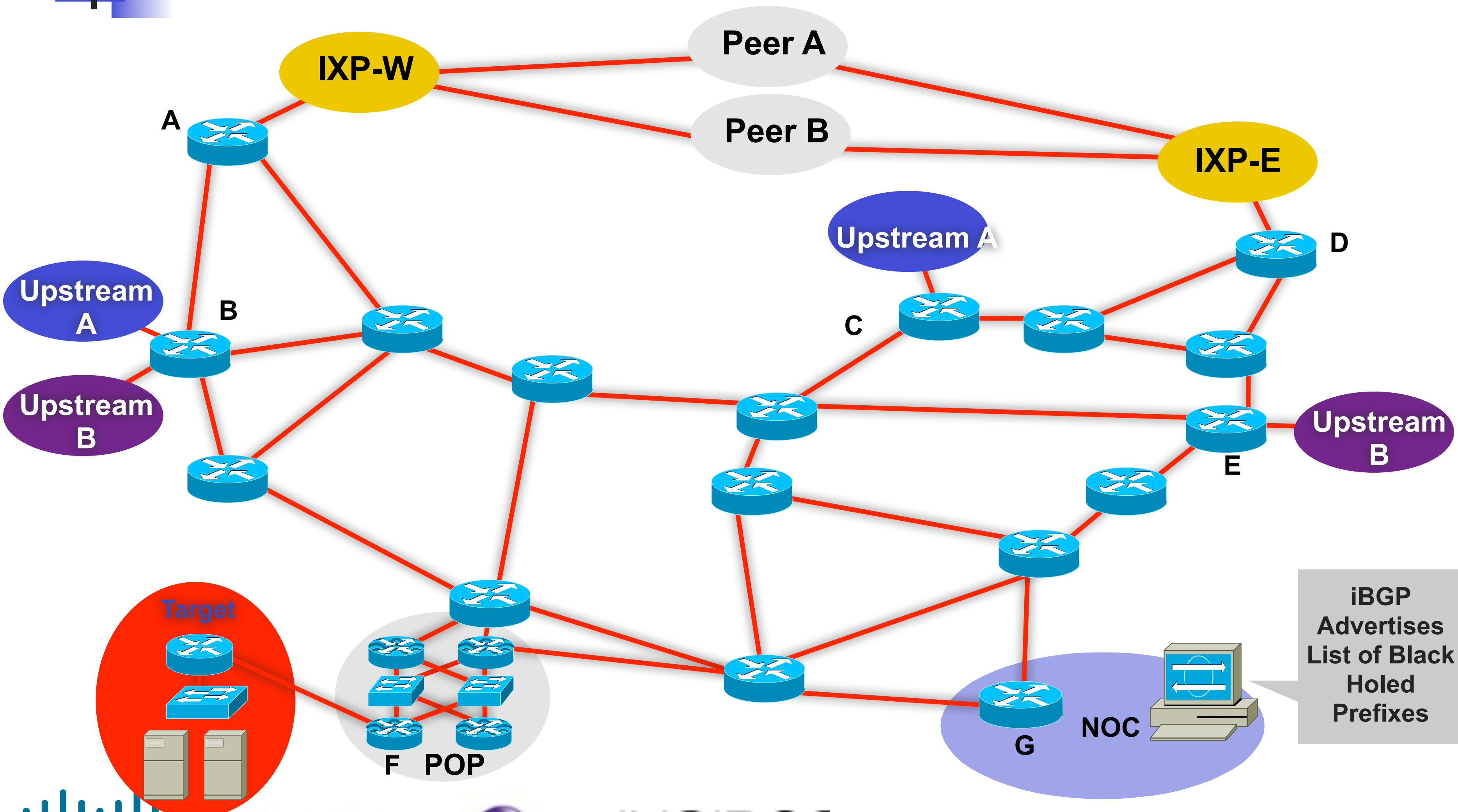
Customer is DOSed – After



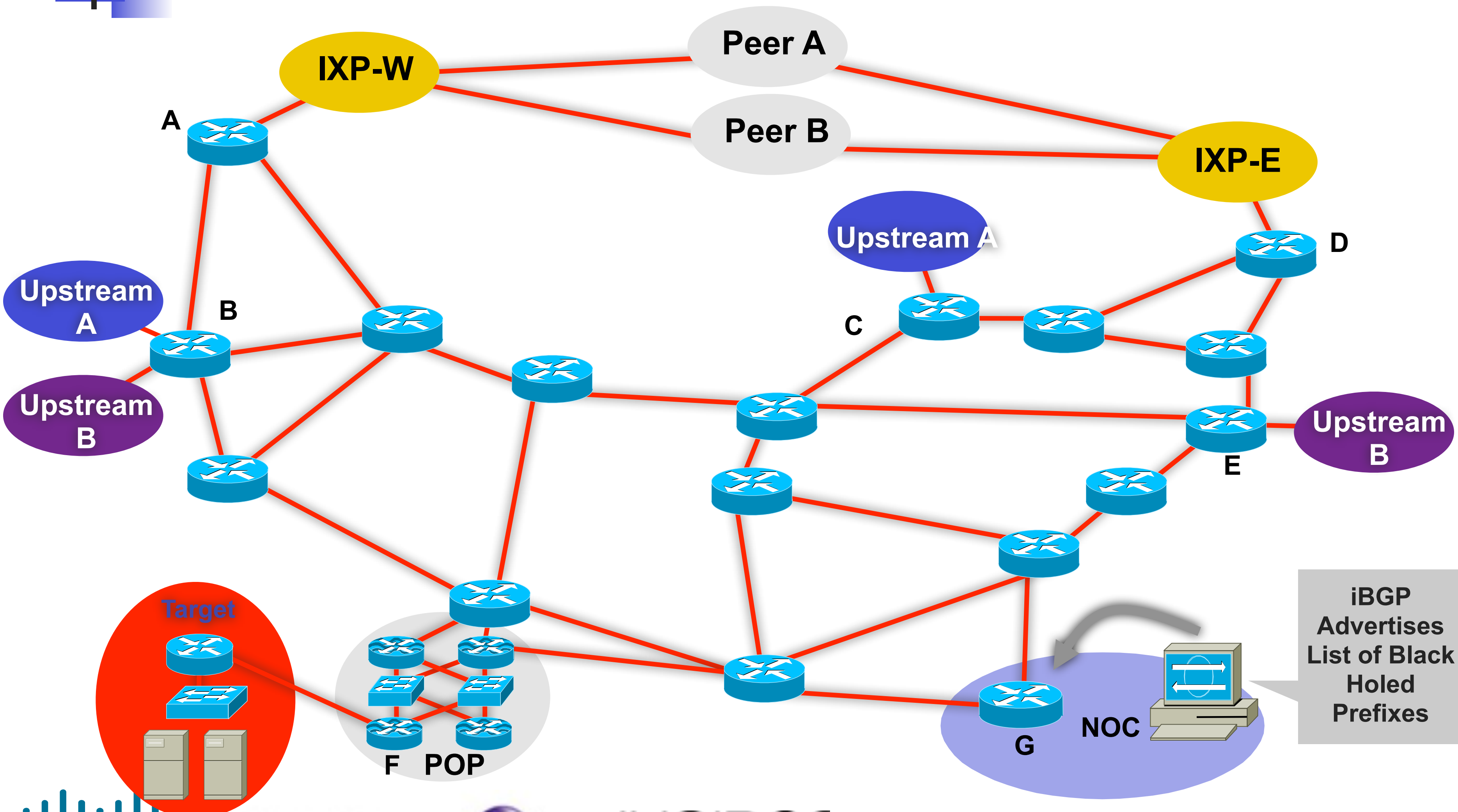
Customer is DOSed – After



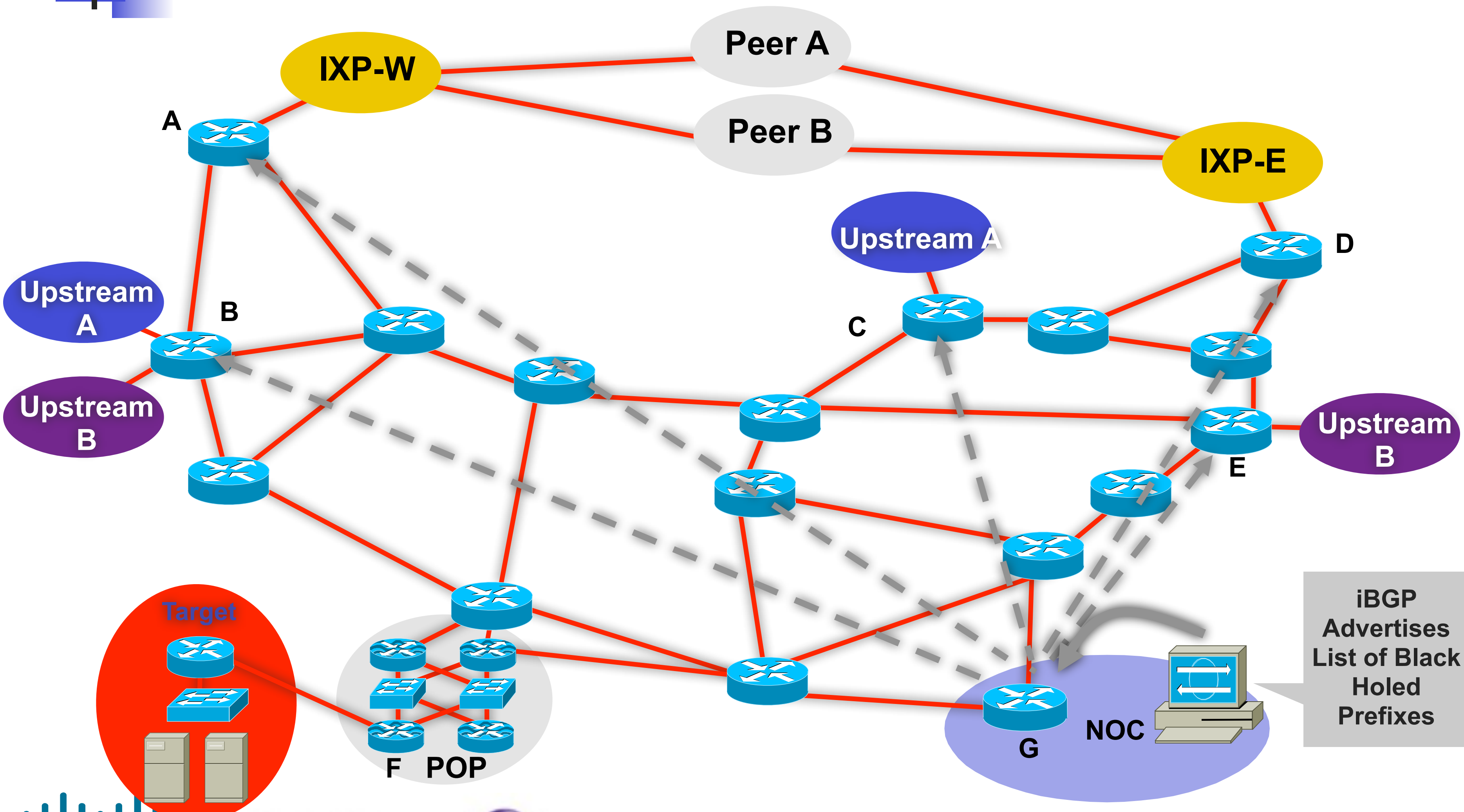
Customer is DOSed – After – Packet Drops Pushed to the Edge



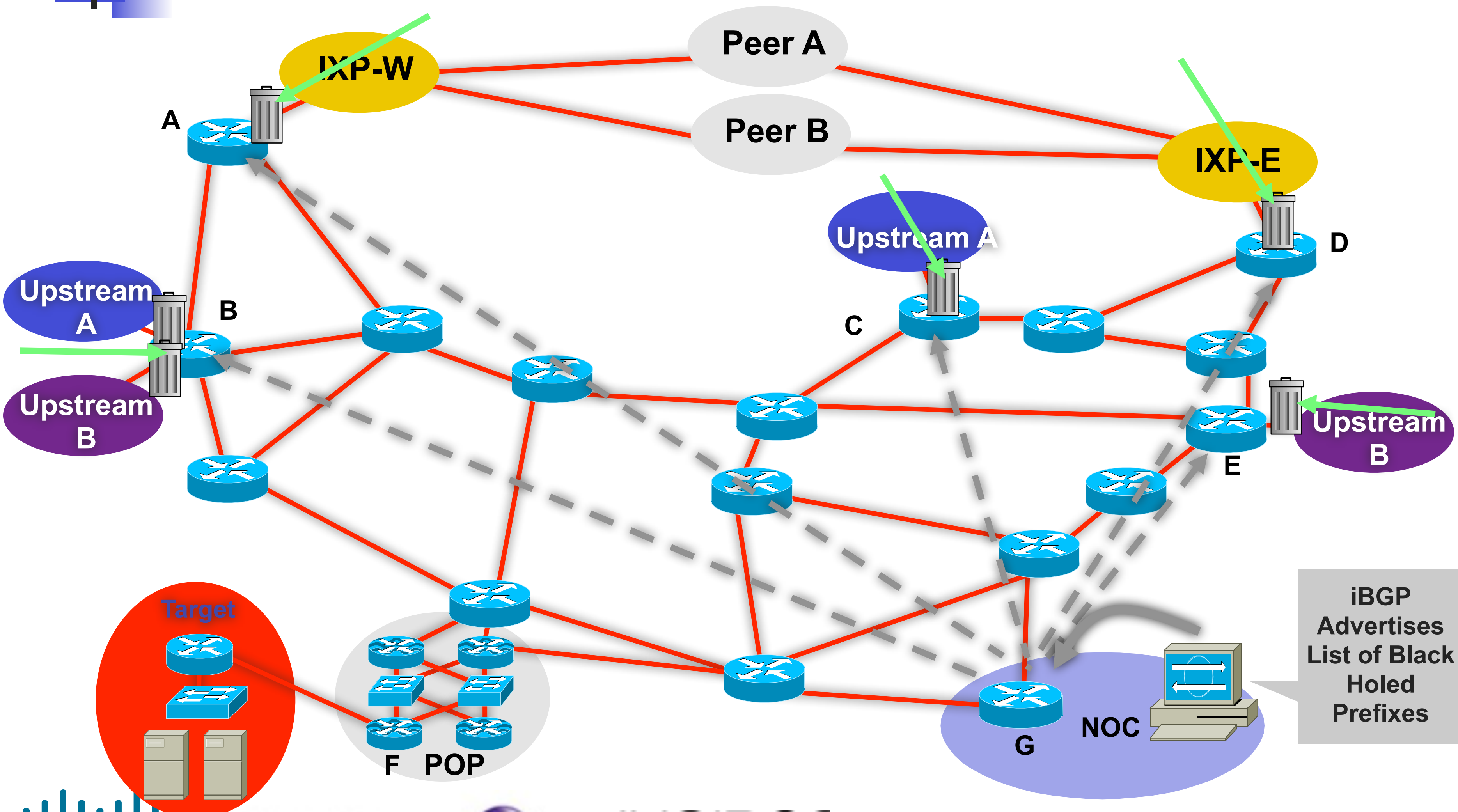
Customer is DOSed – After – Packet Drops Pushed to the Edge



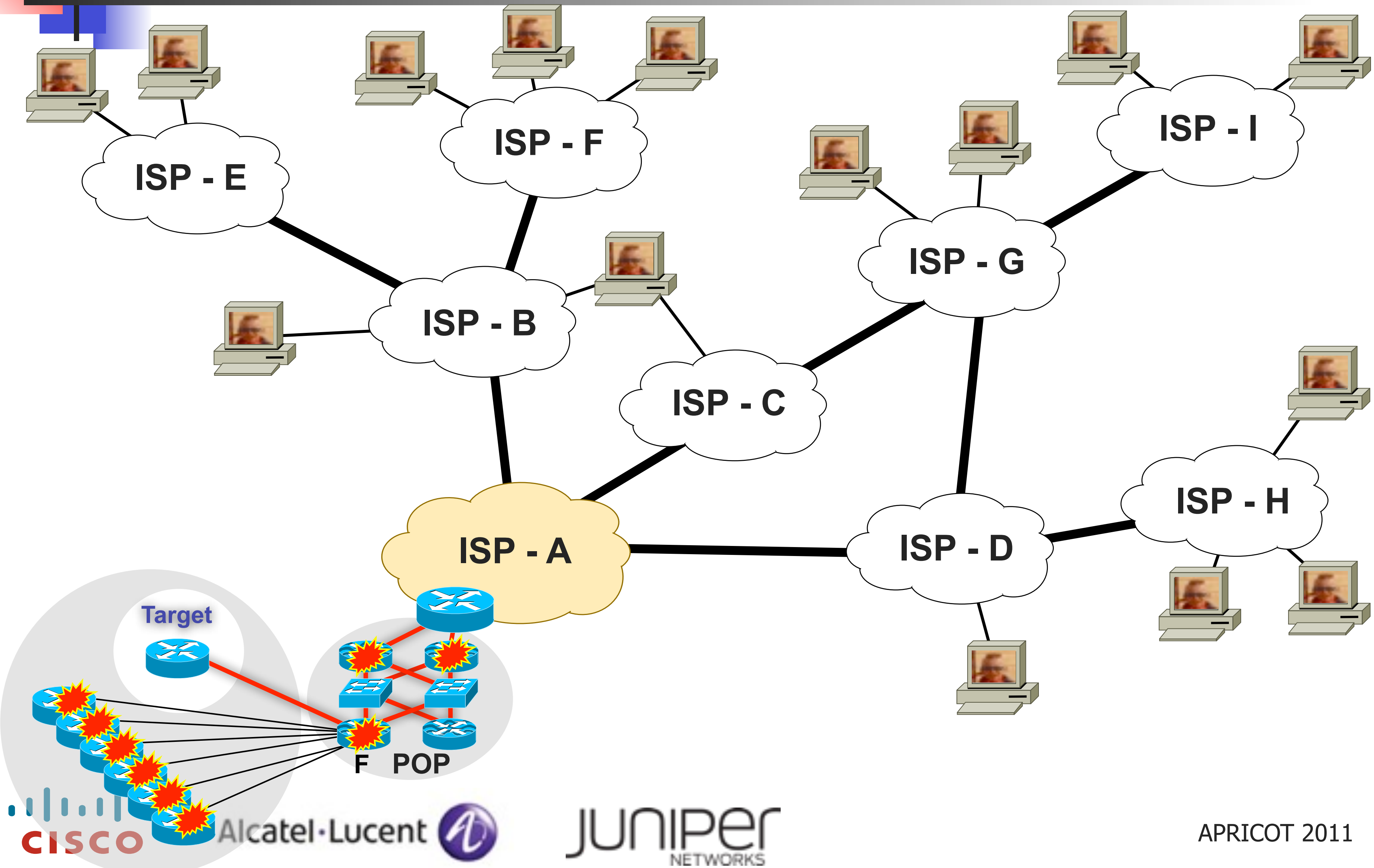
Customer is DOSed – After – Packet Drops Pushed to the Edge



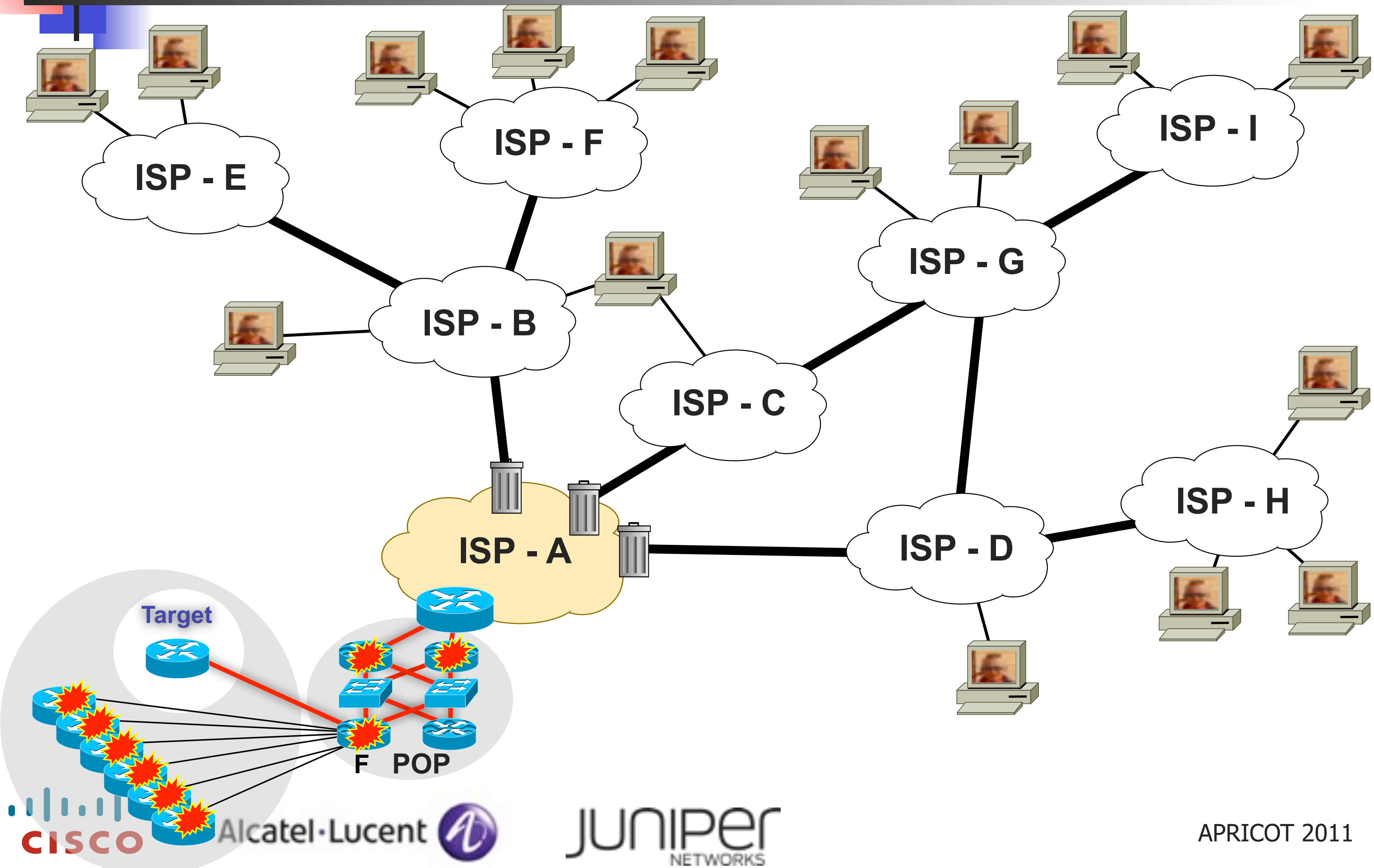
Customer is DOSed – After – Packet Drops Pushed to the Edge



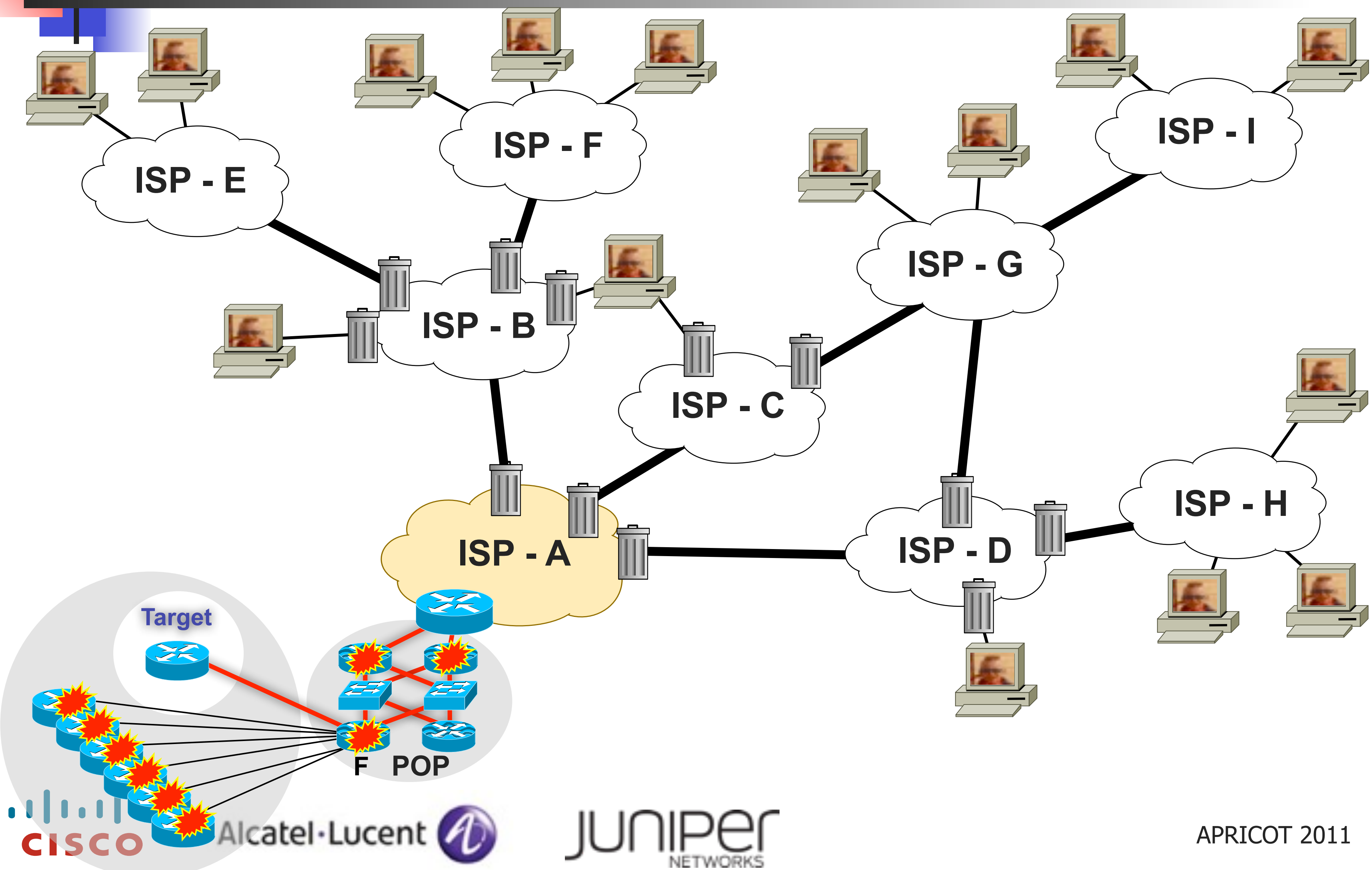
Inter-Provider Mitigation



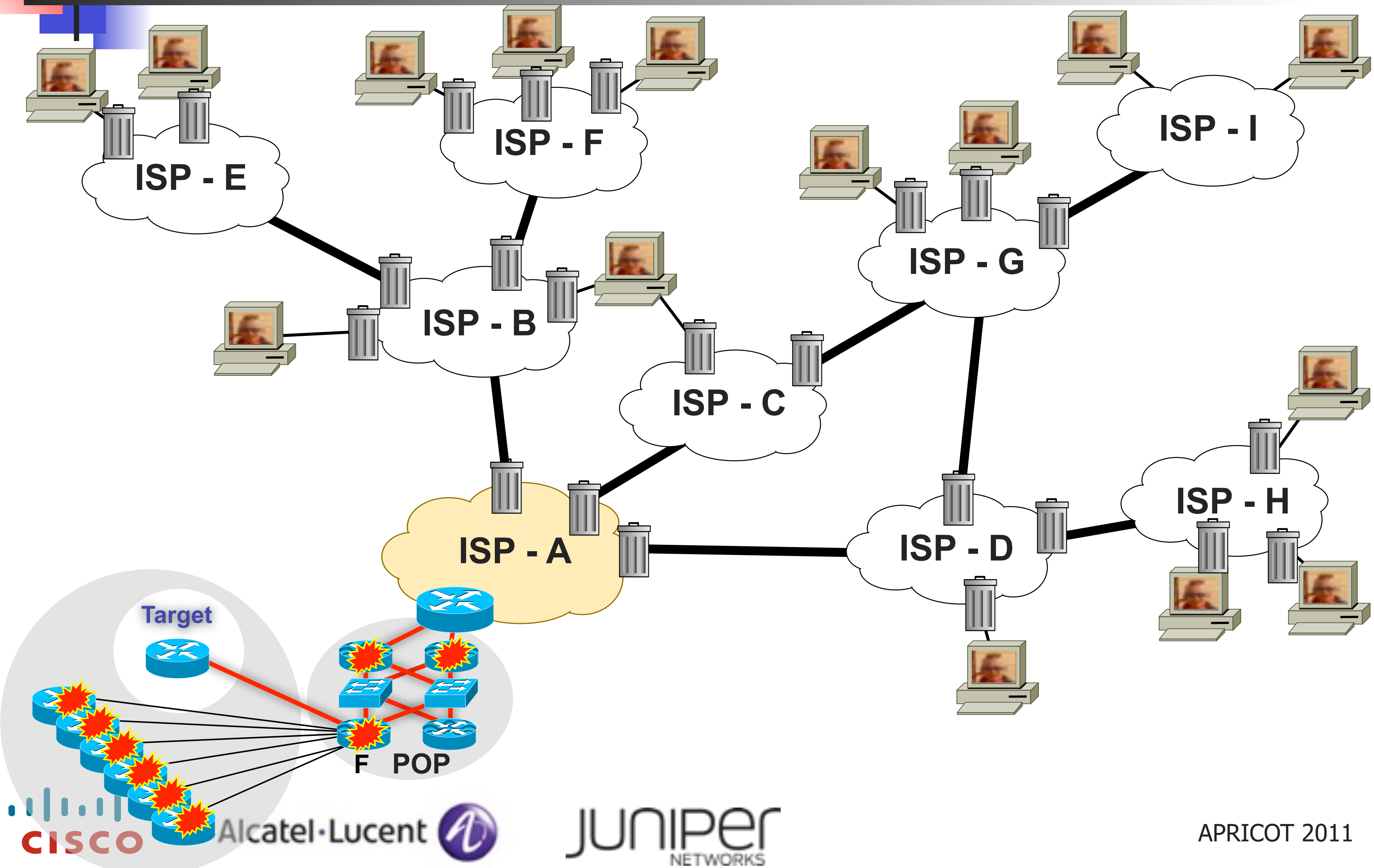
Inter-Provider Mitigation



Inter-Provider Mitigation

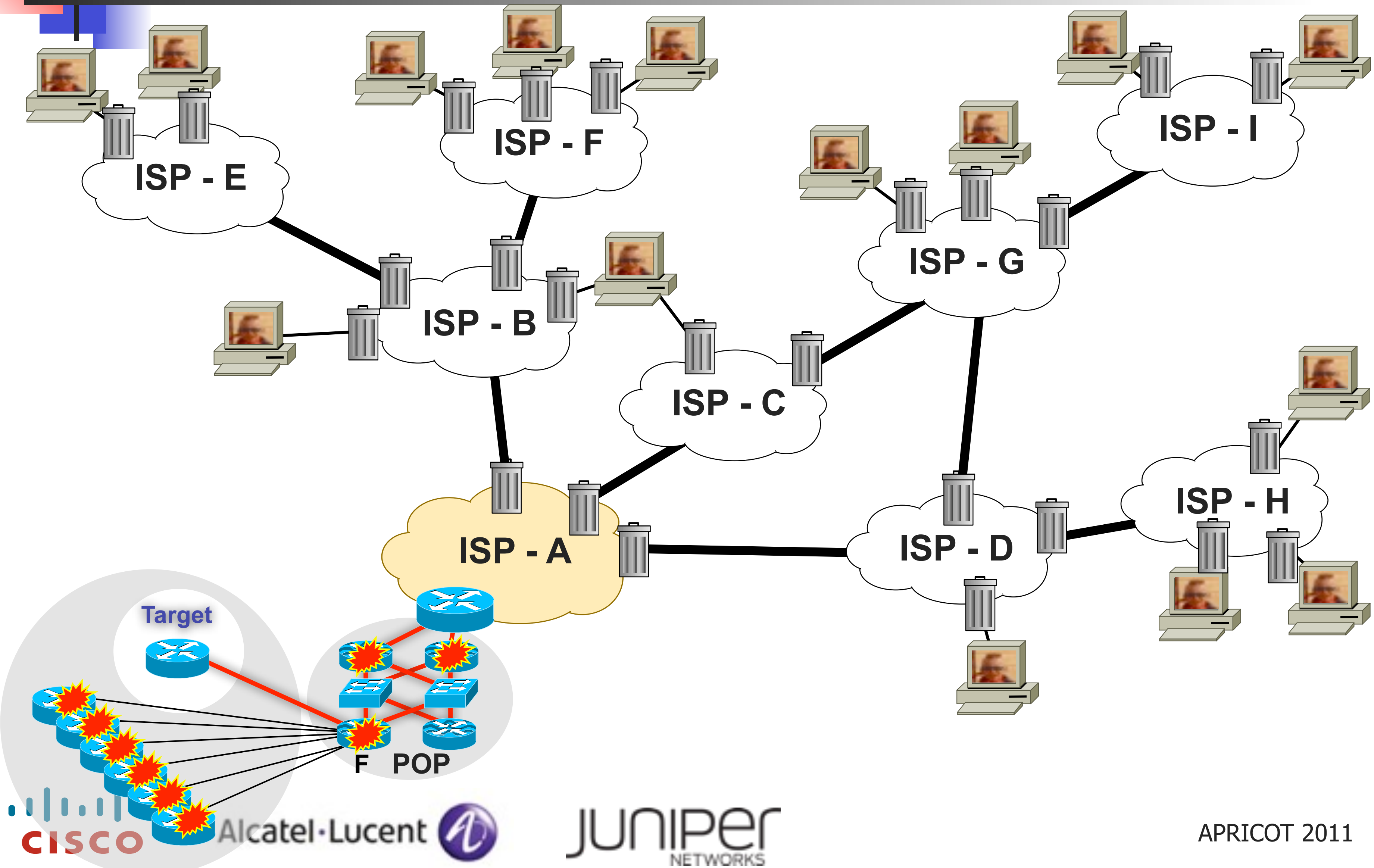


Inter-Provider Mitigation



APRICOT 2011

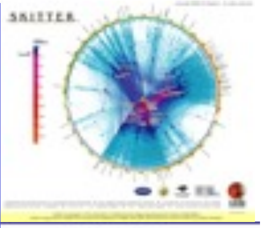
Inter-Provider Mitigation

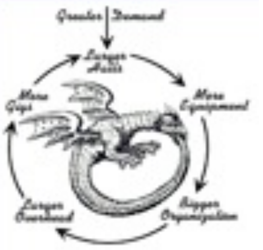


What can you do to help?

- Remote Triggered Black Hole Filtering is the most common ISP DOS/DDOS mitigation tool.
- Prepare your network:
  - ftp://ftp-eng.cisco.com/cons/isp/essentials/ (has whitepaper)
  - ftp://ftp-eng.cisco.com/cons/isp/security/ (has PDF Presentations)
  - NANOG Tutorial:
    - http://www.nanog.org/mtg-0110/greene.html (has public VOD with UUNET)

Sink Holes

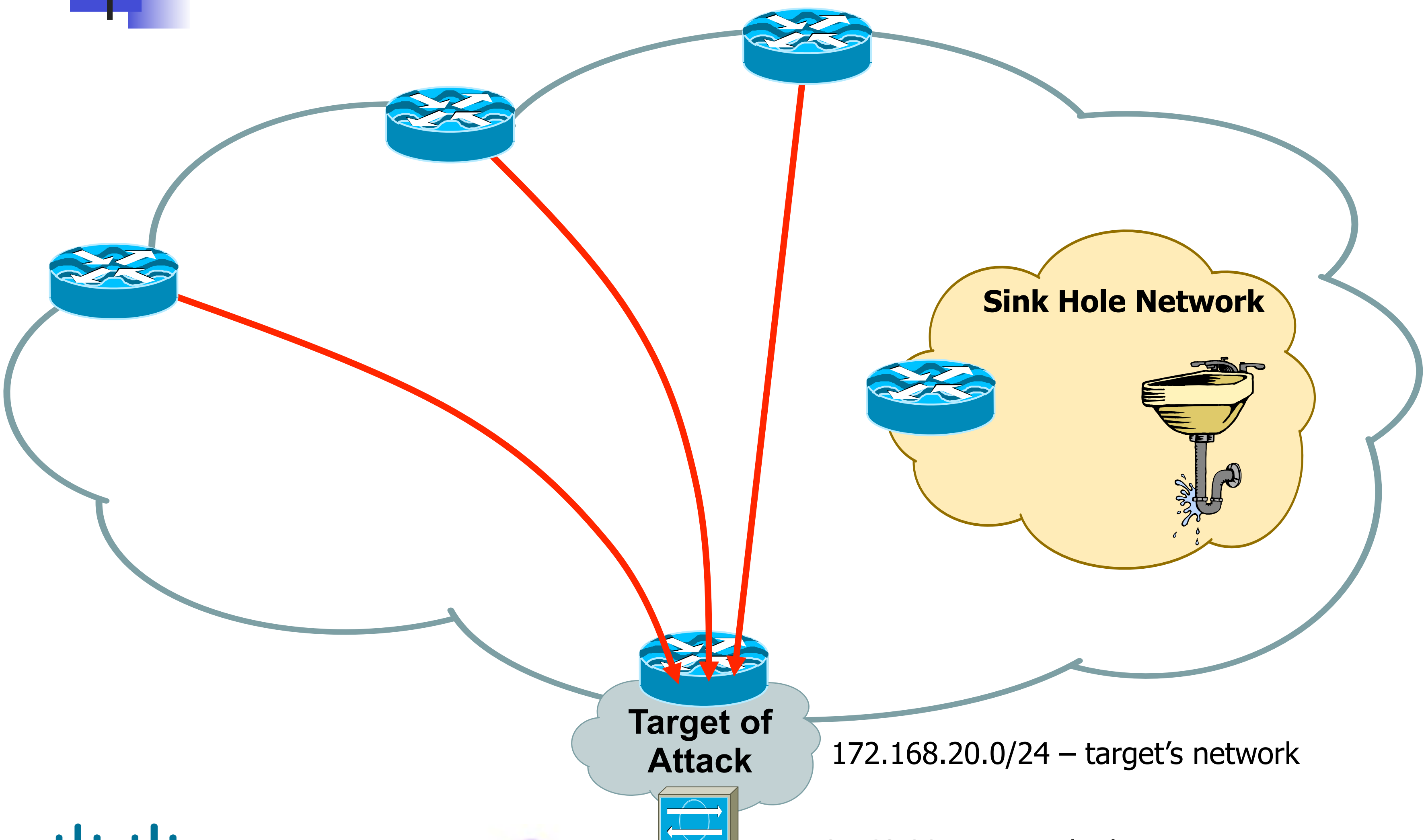

SKITTER


Greater Demand
More Gigs
More Equipment
Larger Overhead
Bigger Organization

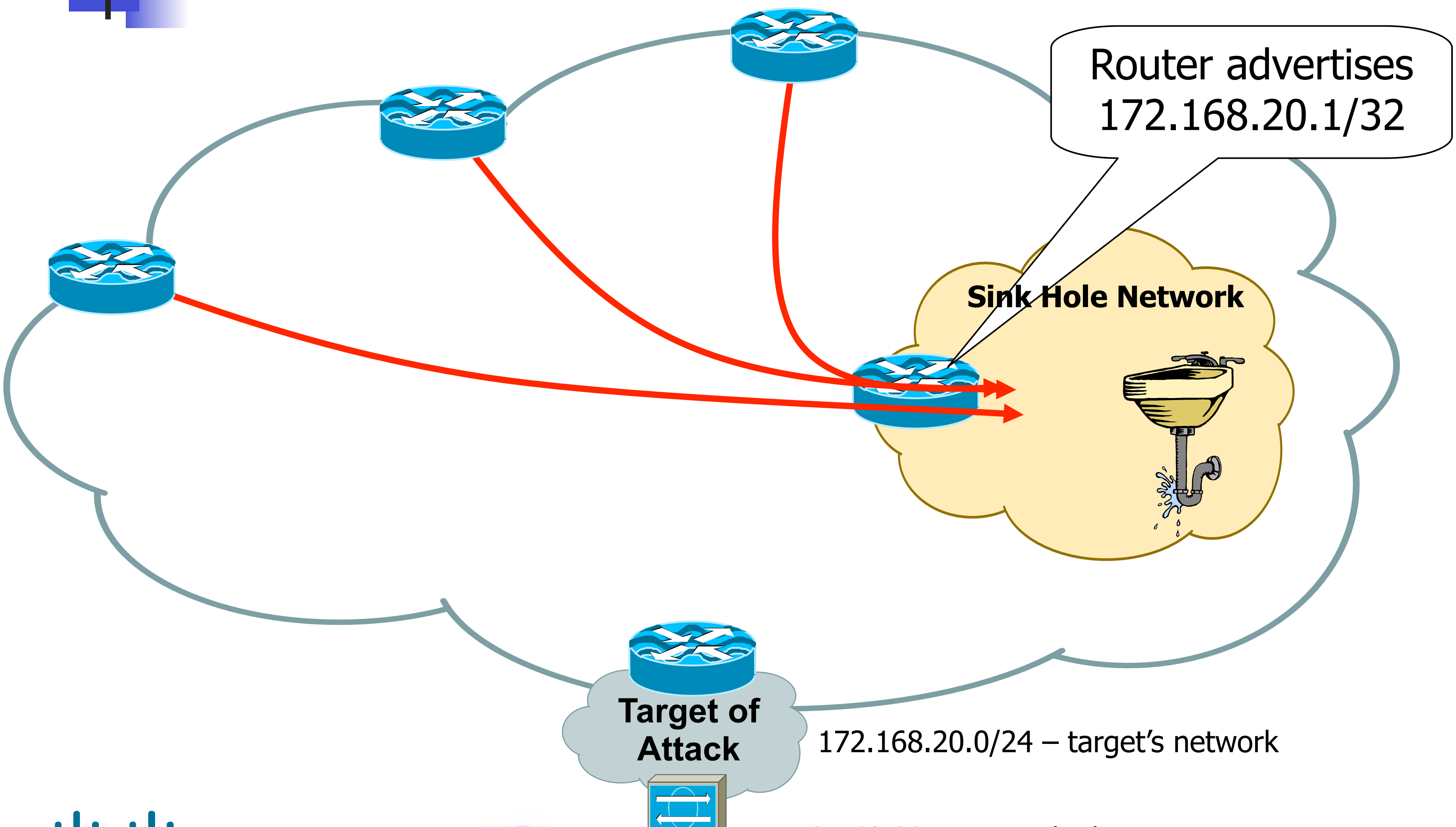

Sink Hole Routers/Networks

- Sink Holes are a Swiss Army Knife security tool.
  - 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)
  - http://www.nanog.org/mtg-0306/sink.html

Sink Hole Routers/Networks

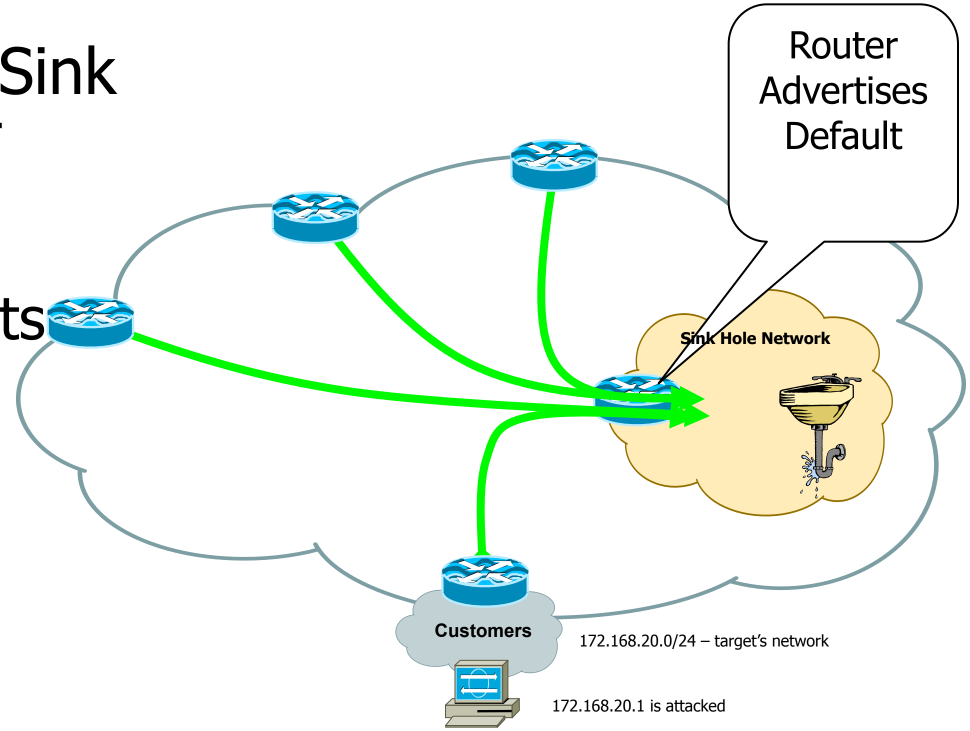


Sink Hole Routers/Networks

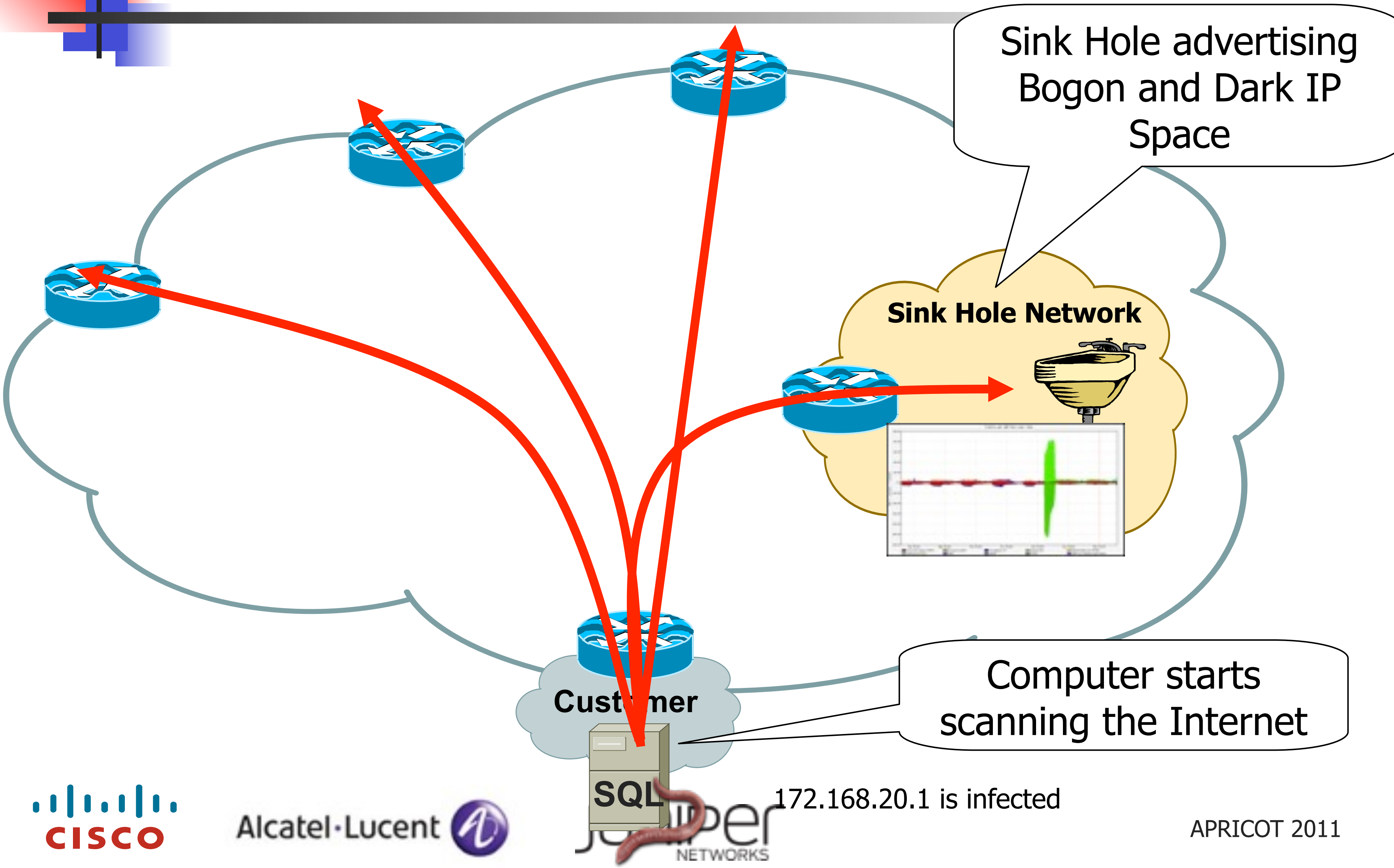


Sink Hole Routers/Networks

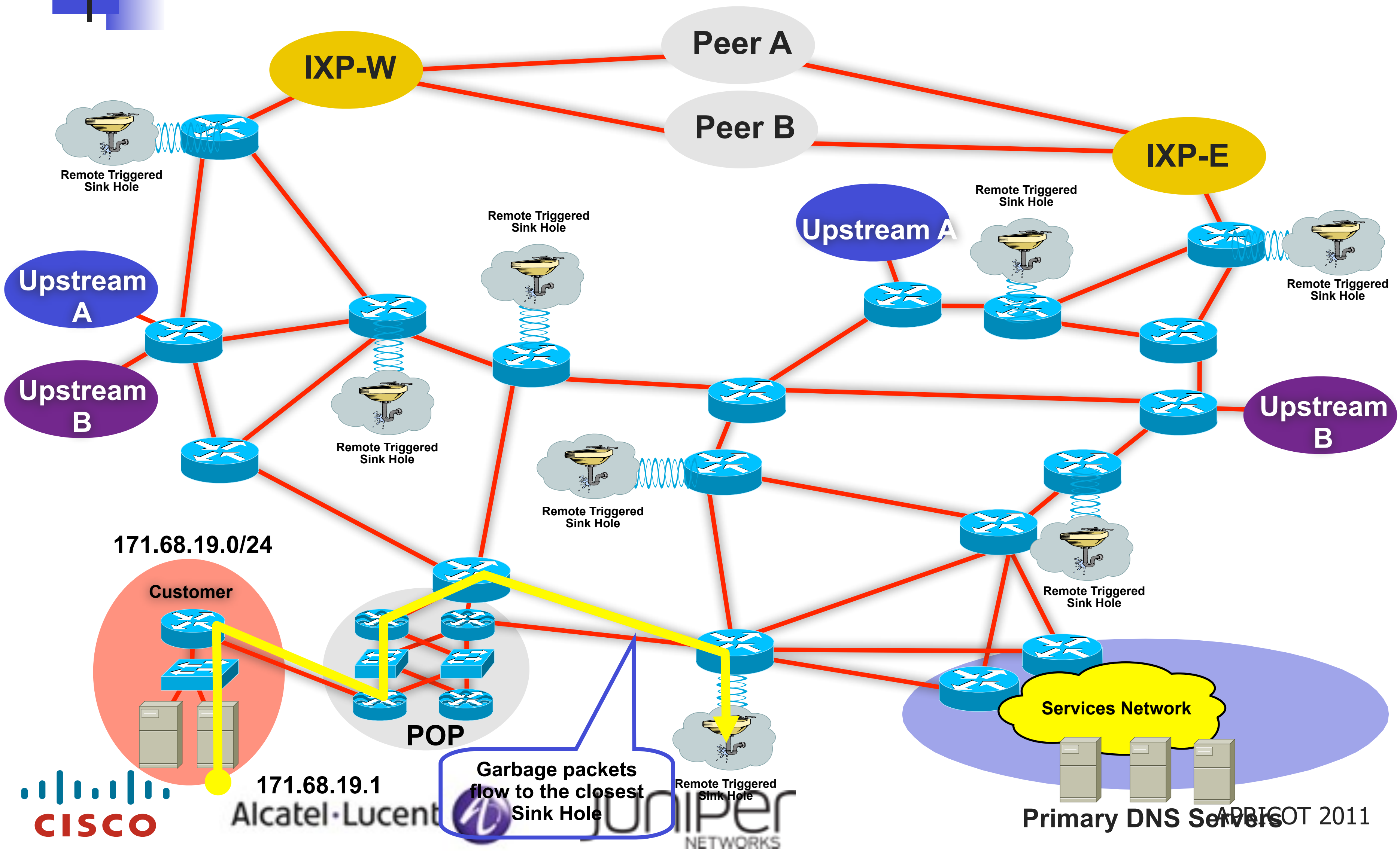
- Advertising Default from the Sink Hole will pull down all sort of junk traffic.
  - Customer Traffic when circuits flap.
  - Network Scans
  - Failed Attacks
  - Code Red/NIMDA
  - Backscatter
- Can place tracking tools and IDA in the Sink Hole network to monitor the noise.



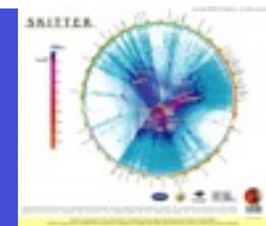
Infected End Points

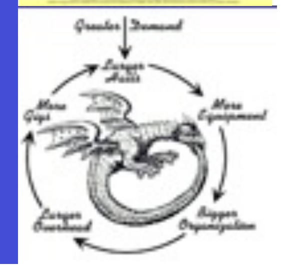
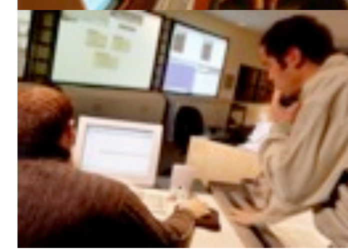


Anycast Sink Holes



Source Address Validation



BCP 38 Ingress Packet Filtering

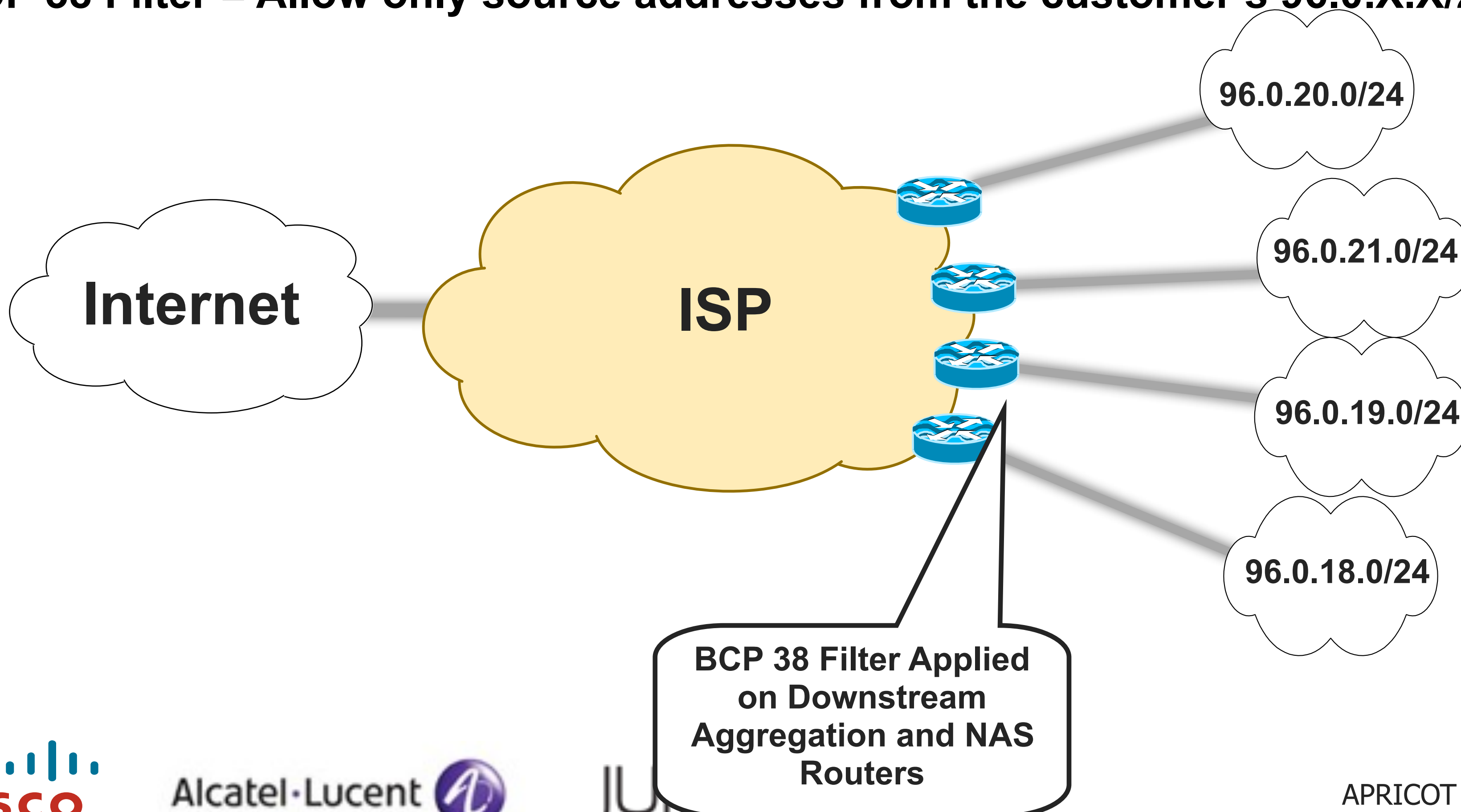
Your customers should not be sending any IP packets out to the Internet with a source address other then the address you have allocated to them!

BCP 38 Ingress Packet Filtering

- BCP 38/ RFC 2827
- Title: Network Ingress Filtering: Defeating Denial of Service Attacks which Employ IP Source Address Spoofing
- Author(s): P. Ferguson, D. Senie

BCP 38 Ingress Packet Filtering

ISP's Customer Allocation Block: 96.0.0.0/19
BCP 38 Filter = Allow only source addresses from the customer's 96.0.X.X/24



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

Many *Working* Techniques

- Static access list on the edge of the network
- Dynamic access list with AAA profiles
- Unicast RPF
- Cable Source Verify (MAC & IP)
- Packet Cable Multimedia (PCMM)
- IP Source Verify (MAC & IP)

Source Address Validation Works

- Successful ISPs have extremely conservative engineering practices.
- Operational Confidence in the equipment, functionality, and features are a prerequisite to any new configs on a router.
- The core reason why ISPs have not been turning on Source Address Validation is their lack of Operational Confidence.

One Major ISP's Example - uRPF

- Month 1 – Cisco Lab Test and Education to help the customer gain confidence in uRPF.
- Month 2 – One port on one router – turning uRPF Strict Mode on a 16xOC3 Engine 2 LC (Cisco 12000)
- Month 3 – One LC on one router – 16xOC3.
- Month 4 – One router all customer facing LCs
- Month 5 – One POP – all customer facing LCs
- Month 6 – Several routers through out the network (other POPs)
- Month 7 – Adopted as standard config for all new customer circuits. Will migrate older customer over time.

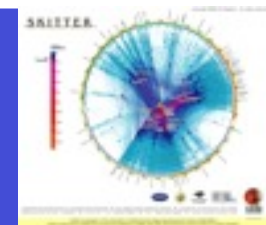
One Major ISP's Example - uRPF

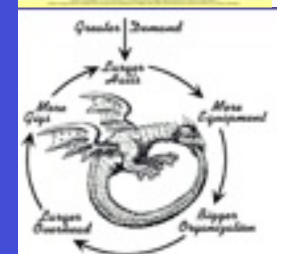
- Lessons Learned:
  - It took time and patience.
  - uRPF did not work for all customers. That is OK, uRPF is not suppose to be a universal solution.
  - Going slow and steady allowed the operations team to gain a feel of the feature's performance envelope --- with out putting the network at risk.
- It works! A year later it is a standard config with over 40K ports running uRPF Strict or Loose Mode.

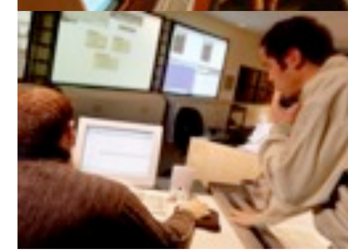
What can you do to help?

- Cut the excuses! BCP 38 is an operational reality!
- Walk them through source address validation techniques, see which ones will work for you, and do not expect more than a 80% success rate.
- Find ways to gain operational confidence in the BCP 38 techniques.
- Source Address validation works – it just take patience and persistence.

Gain Visibility







Total Visibility

- Network Telemetry: Why, What and Where
  - Why does one need to listen to the network?
  - What is one listening to?
  - Where does one gather data or information from?
- Network Telemetry: Tools, Techniques and Protocols
  - How to gather data or information?

Check List

- Check SNMP. Is there more you can do with it to pull down security information?
- Check RMON. Can you use it?
- Check Netflow. Are you using it, can you pull down more?
- See addendum for lots of links.

Review: Six Phases of Incident Response

PREPARATION

Prep the network
Create tools
Test tools
Prep procedures
Train team
Practice

IDENTIFICATION

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

CLASSIFICATION

What kind of attack is it?

TRACEBACK

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

REACTION

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

POST MORTEM

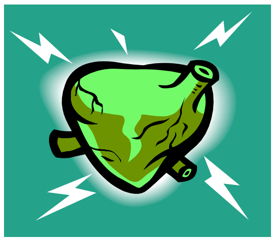
What was done?
Can anything be done to prevent it?
How can it be less painful in the future?

Why Does One Need to Listen to the Network?

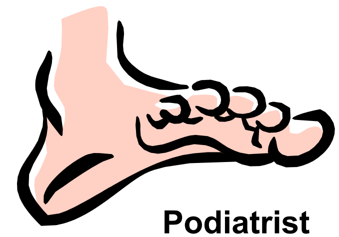
- First and foremost
  - Helps with the other 5 steps [P-I-C-T-R-P]
- Helps to understand the network baseline and behavior
- To understand telemetry elements for information gathering
  - sources (data collection points),
  - protocols to use for data collection
  - telemetry tools
- In the event of security incident
  - to identify and know beforehand that what information is available for forensic work – audit trail
  - faster response time to restore availability when using telemetry during a security incident

Holistic Approach to System-Wide Telemetry

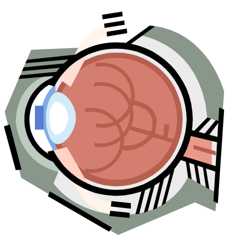
Holistic Approach to System-Wide Telemetry



Cardiologist



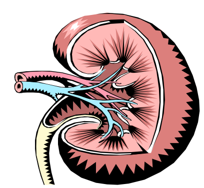
Podiatrist



Ophthalmologist

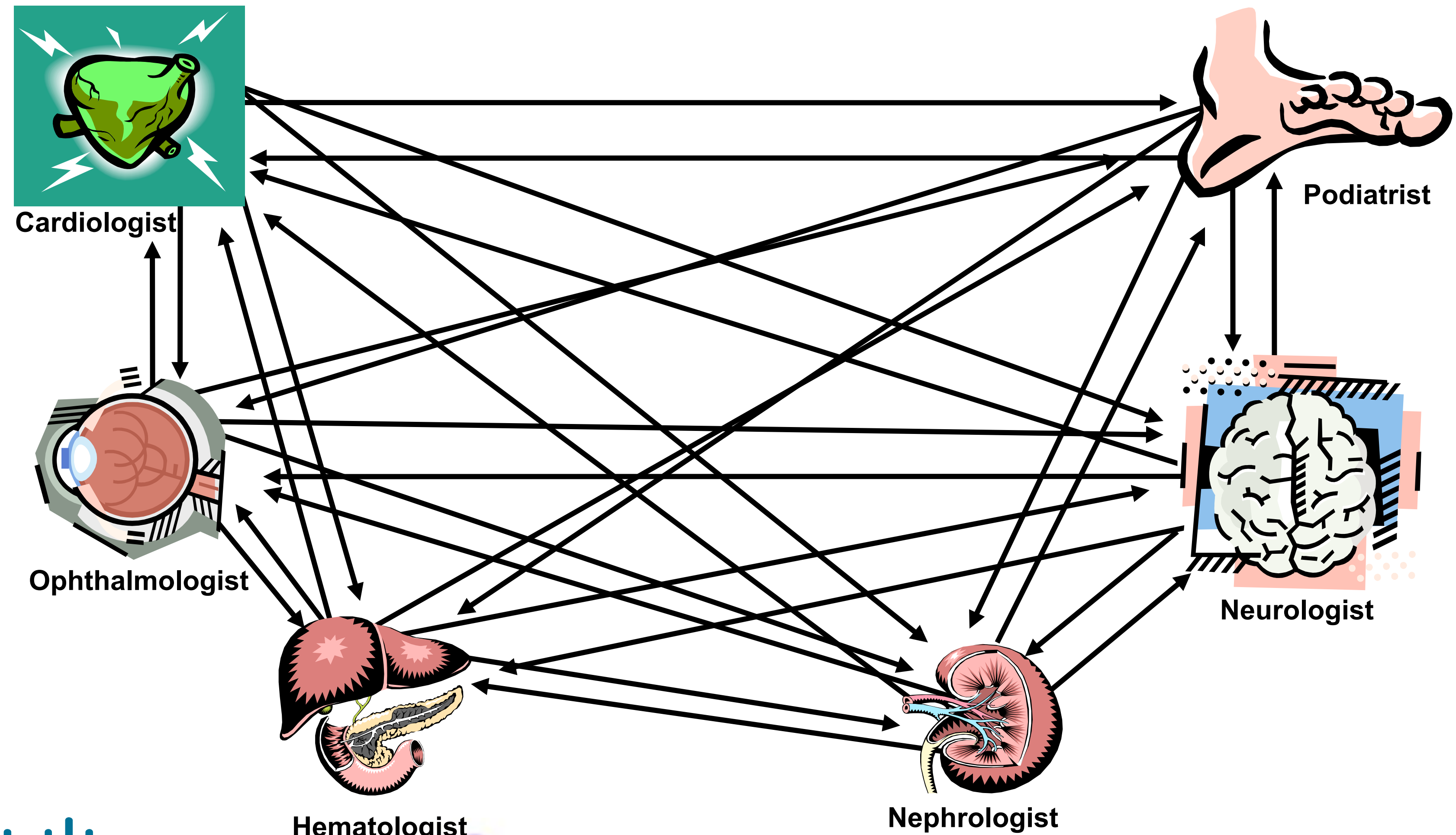
Neurologist

Hematologist

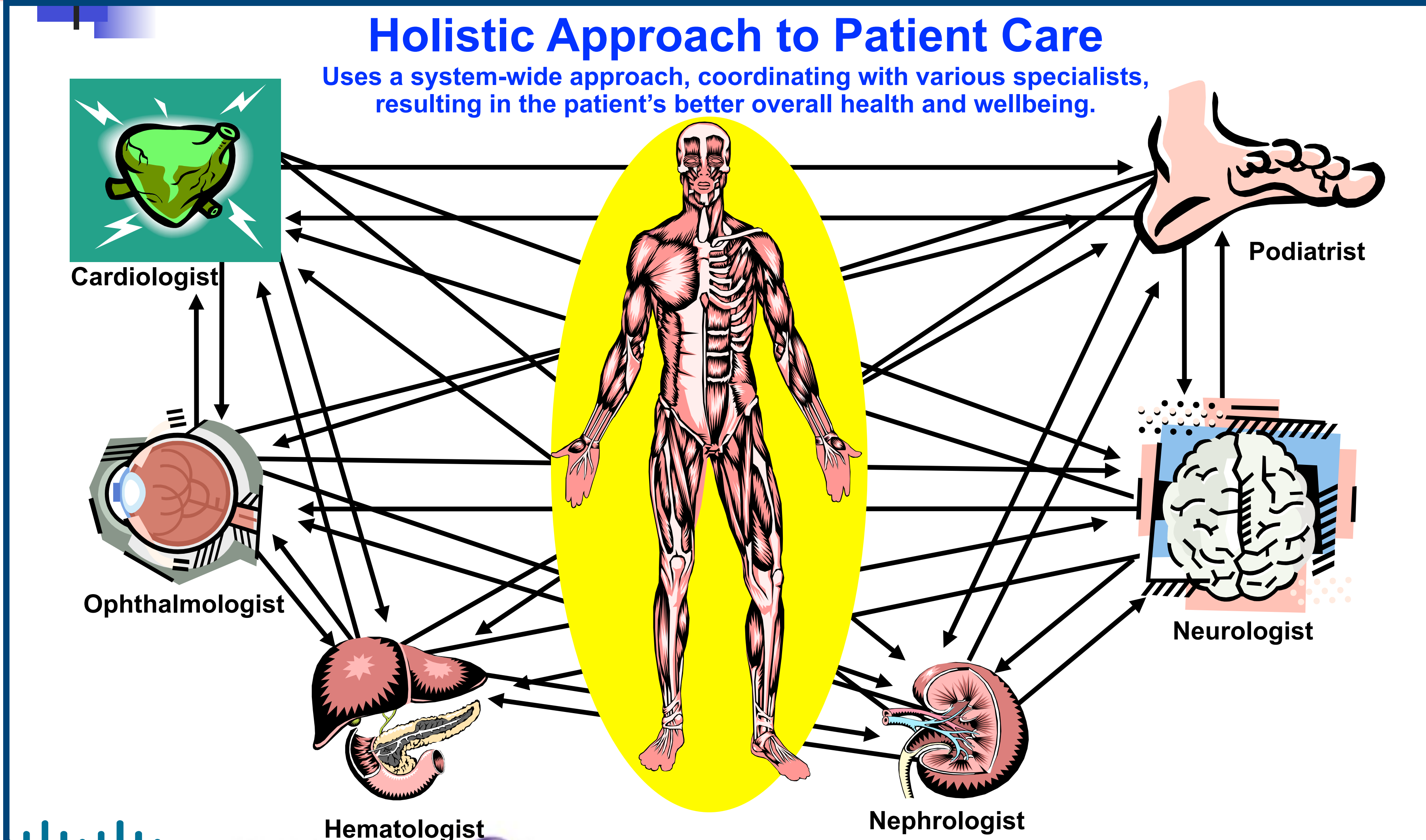


Nephrologist

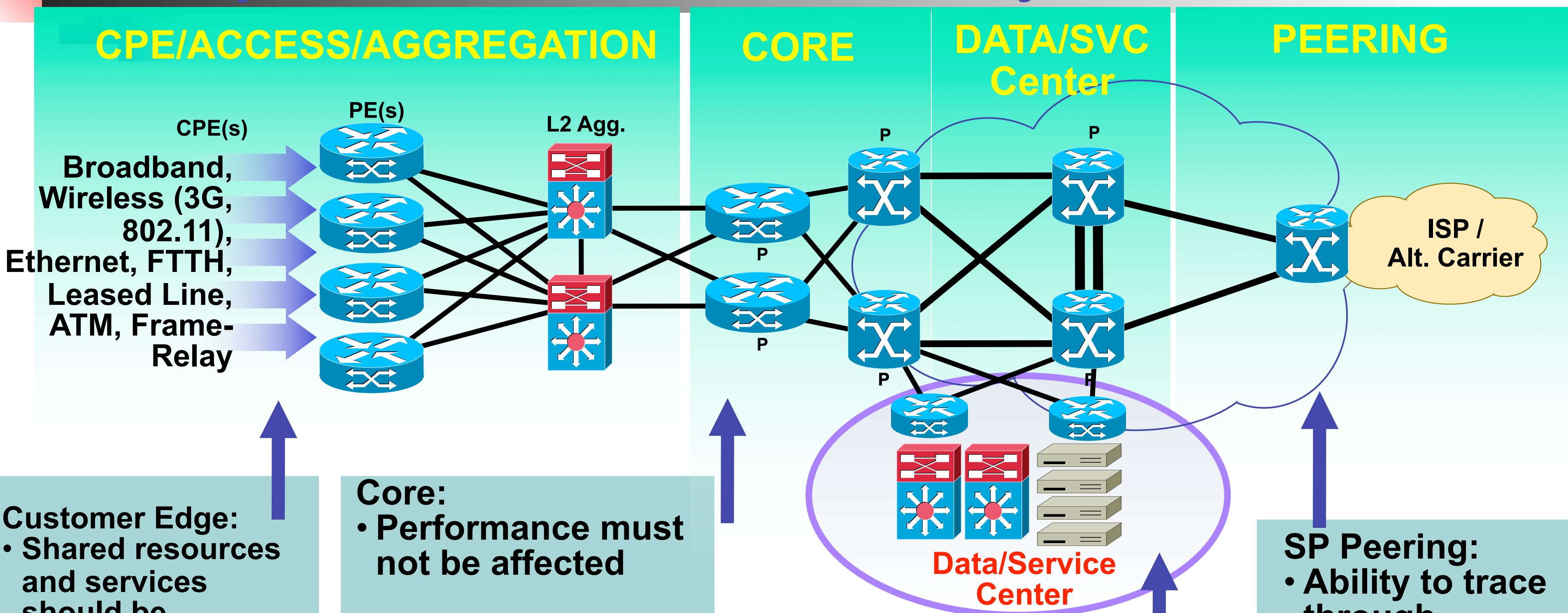
Holistic Approach to System-Wide Telemetry



Holistic Approach to System-Wide Telemetry



Holistic Approach to System-Wide Telemetry



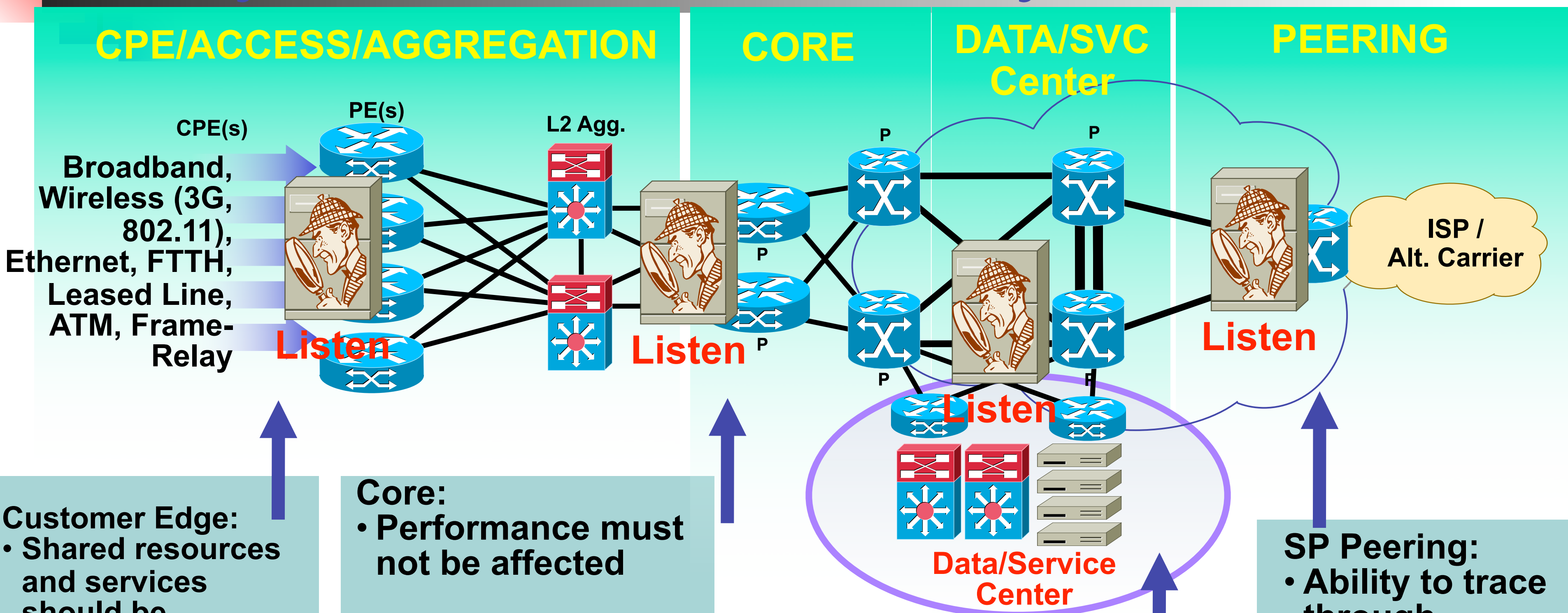
Customer Edge:
- Shared resources and services should be available

Core:
- Performance must not be affected

Data Center:
- Inter as well as Intra Data Center traffic

SP Peering:
- Ability to trace through asymmetric traffic

Holistic Approach to System-Wide Telemetry



Customer Edge:
- Shared resources and services should be available

Core:
- Performance must not be affected

Data Center:
- Inter as well as Intra Data Center traffic

SP Peering:
- Ability to trace through asymmetric traffic

Understand the Concept of Data

Understand the Concept of Data

Risks and threats are NOT prevalent in one place ONLY…

Understand the Concept of Data

Risks and threats are NOT prevalent in one place ONLY…

Need to watch everywhere to avoid being eaten by thousand turkeys...

- Listening to a network element
  - Per device listening
  - Local data provide information about local threats
- Listening to Many
  - Correlation is a MUST
  - Intelligent analysis is a MUST

Listen

High CPU

- Spikes in CPU load on routers, switches, servers, and other devices is often an indication that an event is taking place. Such occurences should always be investigated.
- However, high CPU is not always an indicator of malicious activity. It is important to have both a baseline of historical CPU utilization statistics as well as an understanding of the various processes running on a given system, in order to determine the cause of CPU spikes.
- Correlating CPU utilization with other information such as network traffic statistics, routing-table changes, etc., is often required in order to gain an accurate understanding of the cause(s) and impact of an event.

Link-Flaps

- Link-flaps are also an indication that something is amiss.
- They're often a sign of misconfiguration, backhole incidents and the like - but they can also result from malicious activity, such as a DoS attack against a router causing a reload due to CPU spike, and hence a link-flap.
- Routers and switches can be configured to notify monitoring systems when link-flaps occur.
- Correlating link-flaps with other forms of information is often necessary in order to gain a complete understanding of an event.

Instrumentation

- Network instrumentation offers the most extensive and useful detection capabilities.
- This instrumentation is often coupled with dedicated analysis systems which collect, analyze, and correlate information from disparate sources in order to present a more complete view of events taking place within the network.
- There are several forms of instrumentation built into routers, switches, and other network devices. Instrumentation is also present in most modern general-purpose operating systems.
- There are a number of open source and commercial tools available which greatly enhance the utility of instrumentation.
- Getting started with network instrumentation is both inexpensive and relatively easy.

Example - sh proc c

7600>show proc c | e 0.00%__0.00%__0.00%

```
CPU utilization for five seconds: 38%/26%; one minute: 40%; five minutes: 43%
 PID Runtime(ms)    Invoked      uSecs    5Sec   1Min   5Min TTY Process
   5   192962596   13452649      14343  0.00%  0.52%  0.44%   0 Check heaps
  15   4227662201540855414         274  0.65%  0.50%  0.49%   0 ARP Input
  26   262901268368047372 6         71  0.24%  0.29%  0.36%   0 Net Background
  50     9564564   11374799        840  0.08%  0.07%  0.08%   0 Compute load avg
  51    15291660     947844      16133  0.00%  0.03%  0.00%   0 Per-minute Jobs
  58    15336356   92241638        166  0.08%  0.02%  0.00%   0 esw_vlan_stat_pr
  67    10760516  506893631         21  0.00%  0.01%  0.00%   0 Spanning Tree
  68  3180465968 2556402094       1244  7.02%  7.04%  7.75%   0 IP Input
  69    25488912   65260648        390  0.00%  0.03%  0.00%   0 CDP Protocol
  73    16425564   11367610       1444  0.08%  0.02%  0.00%   0 QOS Stats Export
  81    12460616    1020497      12210  0.00%  0.02%  0.00%   0 Adj Manager
  82   442430400   87286325       5068  0.65%  0.73%  0.74%   0 CEF process
  83    68812944   11509863       5978  0.00%  0.09%  0.11%   0 IPC LC Message H
  95    54354632   98373054        552  0.16%  0.12%  0.13%   0 DHCPD Receive
  96    61891604   58317134       1061  1.47%  0.00%  4.43%   0 Feature Manager
 111        9420      12010        784  0.00%  0.23%  0.46%   0 Exec
 165   1817346481141817381         159  0.32%  0.57%  0.40%   0 IP SNMP
 166   117953648  573360040        205  0.00%  0.32%  0.26%   0 PDU DISPATCHER
```

CLI
Pipes

Example - sh proc c

```
7600>sh proc c | e 0.00
CPU utilization for five seconds: 41%/26%; one minute: 46%; five minutes: 44%
 PID Runtime(ms)    Invoked      uSecs    5Sec    1Min    5Min TTY Process
  15    4227657321540854233        274   0.40%   0.39%   0.47%   0 ARP Input
  26    2629008963680468704         71   0.08%   0.36%   0.39%   0 Net Background
  50      9564512   11374786       840   0.08%   0.07%   0.08%   0 Compute load avg
  68   3180457804255618343 0      1244   9.65%   8.49%   7.75%   0 IP Input
  69     25488888   65260576       390   0.32%   0.05%   0.01%   0 CDP Protocol
  82    442429604   87286223      5068   0.73%   0.73%   0.74%   0 CEF process
 :
 175          624         92      6782   0.57%   0.49%   0.16%   1 SSH Process
```

CLI Pipes allow clean and crisp output

IOS CLI - sh proc c (cont.)

- There are processes which are platform-specific - i.e., `Feature Manager` is found on the 6500/7600 only, while `IPC CBus` is 7500-specific.
- Aliasing the more complex sh proc c commands to a single-letter alias as part of the standard config is extremely useful when the box is under high load and it’s hard to type on the console:

```
Router(config)#alias exec p show proc c |
e 0.00%__0.00%__0.00%
```

- Understanding your platform(s), and what’s normal - including periodically-run processes (`BGP Scanner`, for example) - is key
- On the 12000, one must either `attach` to a linecard or perform an `execute` command specifying a linecard in order to see its CPU load; on the 7500, one uses the `if`

IOS CLI - sh int

- Sh int displays interface-level statistics, including throughput (pps) and bandwidth (bps)
- Typically, routers are set to use a 5-minute decaying average for interface statistics by default - changing this to 1 minute gives more granular statistics
- Looking for high input/output rates over a period of a minute or so can be very helpful
- Clear the counters first, otherwise it's much harder to determine which interfaces are receiving high rates of traffic

Example - sh int

```
GigabitEthernet3/13 is up, line protocol is up (connected)
  Hardware is C6k 1000Mb 802.3, address is 00d0.0136.000a (bia 00d0.0136.000a)
  Description: IP TELEPHONY
  Internet address is 10.89.254.130/26
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex mode, link type is autonegotiation, media type is SX
  output flow-control is unsupported, input flow-control is unsupported, 1000Mb/s
  Clock mode is auto
  input flow-control is off, output flow-control is off
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:00, output 00:00:00, output hang never
  Last clearing of "show interface" counters 1y39w
  Input queue: 0/75/15005/235 (size/max/drops/flushes); Total output drops: 1
  Queueing strategy: fifo
  Output queue :0/40 (size/max)
  5 minute input rate 4751000 bits/sec, 3006 packets/sec
  5 minute output rate 4499000 bits/sec, 2755 packets/sec
  L2 Switched: ucast: 19841909032 pkt, 3347755205145 bytes - mcast: 96885779 pkt, 5131184435 bytes
  L3 in Switched: ucast: 27282638229 pkt, 5095662463006 bytes - mcast: 94 pkt, 5191 bytes mcast
  L3 out Switched: ucast: 43107617667 pkt, 7275264441541 bytes
     47118207406 packets input, 9306459456266 bytes, 0 no buffer
```

Example - sh int

```
12000>sh int po1/1/0 | i 1 minute
  1 minute input rate 56616000 bits/sec, 18097
  packets/sec
  1 minute output rate 120609000 bits/sec, 24120
  packets/sec
12000>sh int po1/1/0 | i 1 minute
  1 minute input rate 59030000 bits/sec, 19171
  packets/sec
  1 minute output rate 111233000 bits/sec, 22365
  packets/sec
12000>sh int po1/1/0 | i 1 minute
  1 minute input rate 54307000 bits/sec, 17637
  packets/sec
  1 minute output rate 119223000 bits/sec, 23936
  packets/sec
```

Rate Interval

IOS CLI - sh ip int

- `sh ip int` gives information about features configured on an interface
- It's useful to get the number or name of an ACL in order to check ACL counter hits (6500/7600 only shows ACL counters on Sup720 w/PFC3BXL)
- uRPF drop information is also available via `sh ip int`, shows information about spoofed and/or RTBH-dropped packets

Example - sh ip int

```
12000>sh ip int po1/1/0 | i veri
  IP verify source reachable-via ANY
   794407 verification drops
   1874428129 suppressed verification
 drops
12000>sh ip int po1/1/0 | i veri
  IP verify source reachable-via ANY
   794408 verification drops
   1874444463 suppressed verification
 drops
```

IOS CLI - sh ip traffic

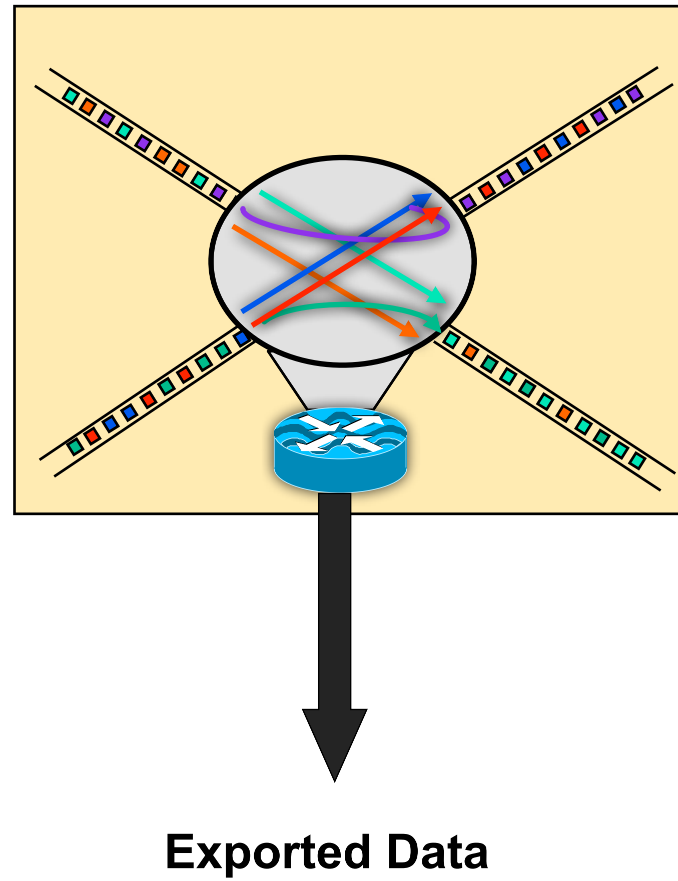
- Sh ip traffic provides a lot of useful global statistics, including per-protocol counts for ICMP, TCP, UDP, and multicast traffic
- Very useful for troubleshooting in general, as well as for spotting oddities
- Also shows global uRPF drop statistics

Example - sh ip traffic

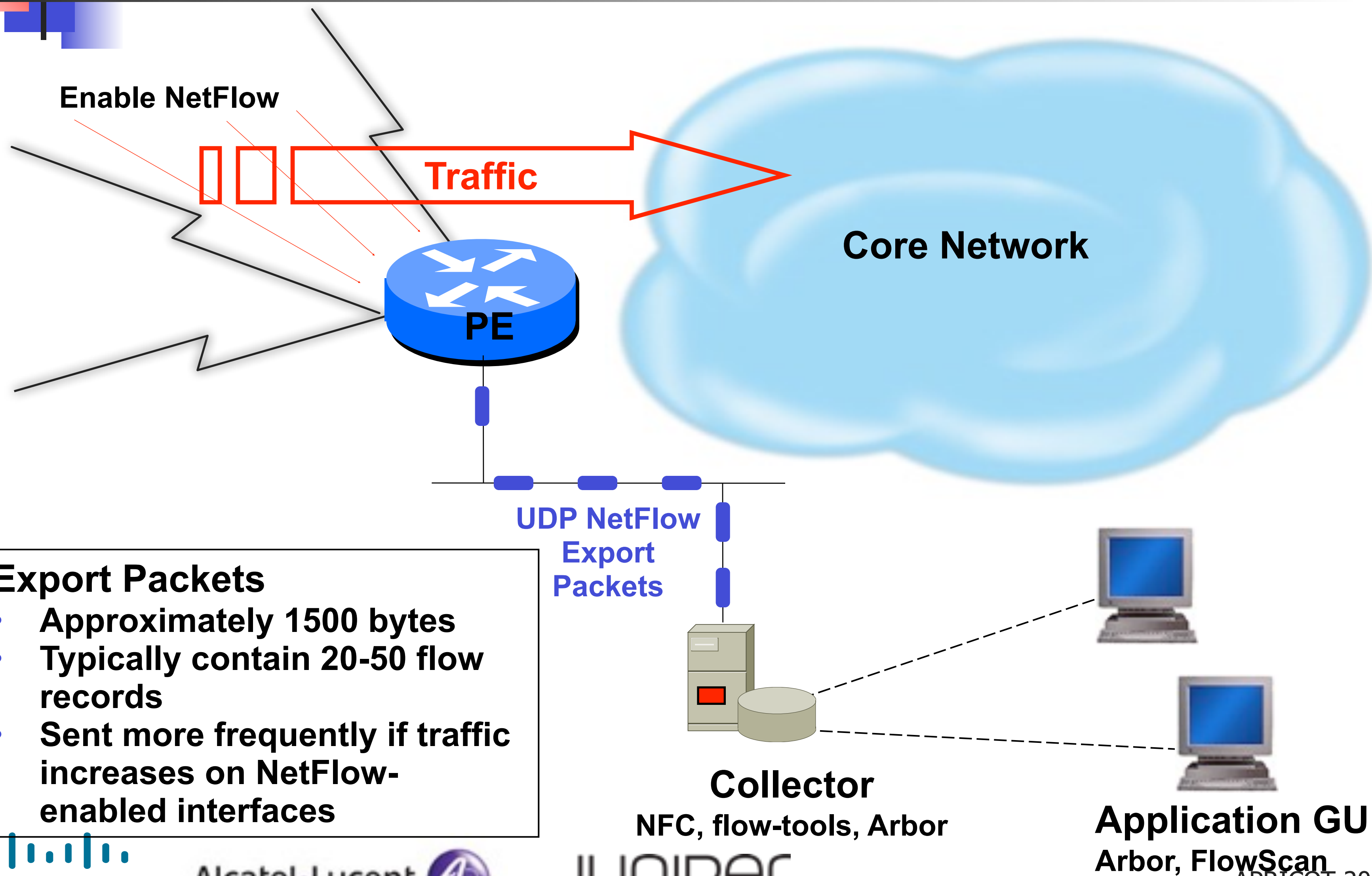
```
12000>sh ip traff | i RPF
        0 no route, 124780722 unicast RPF, 0 forced drop
12000>sh ip traff | i RPF
        0 no route, 124816525 unicast RPF, 0 forced drop
12000>sh ip traff | i RPF
        0 no route, 127777619 unicast RPF, 0 forced drop
12000>sh ip traff | i RPF
        0 no route, 135875095 unicast RPF, 0 forced drop
12000>sh ip traff | i RPF
        0 no route, 150883277 unicast RPF, 0 forced drop
```

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)



Creating Export Packets



Export Packets

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

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 & 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: Clarifying the Version Myth

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 & TOS information
8	Choice of eleven aggregation schemes Reduces resource usage
9	Flexible, extensible file export format to enable easier support of additional fields & technologies; coming out now are MPLS, Multicast, & BGP Next-Hop

Cisco Catalyst 6500 Series Router supports versions 5 and 8 in Cisco IOS Software Release 12.1(13)E

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

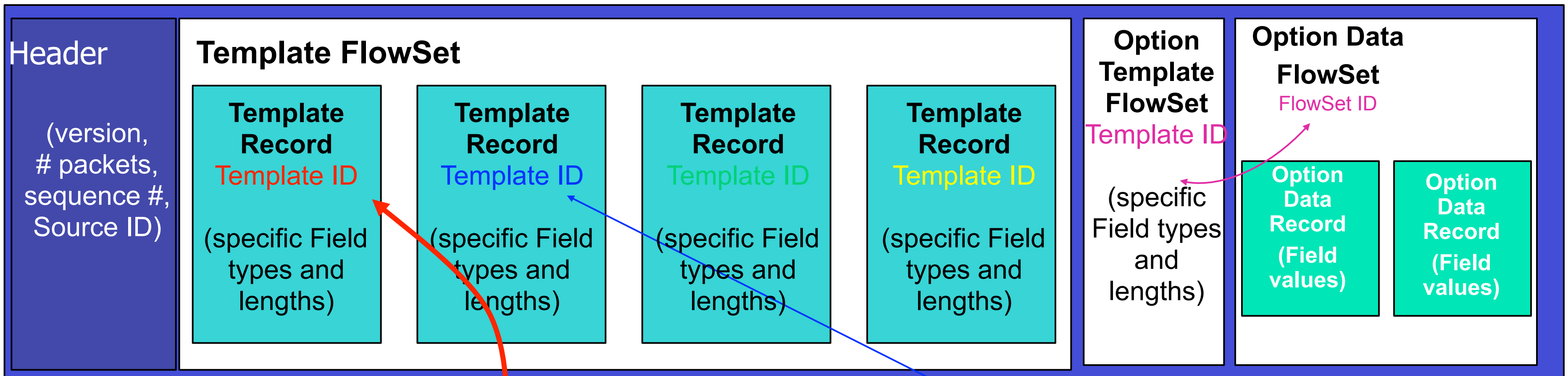
Solution: Build a flexible and extensible export format!

Netflow v9 Principles

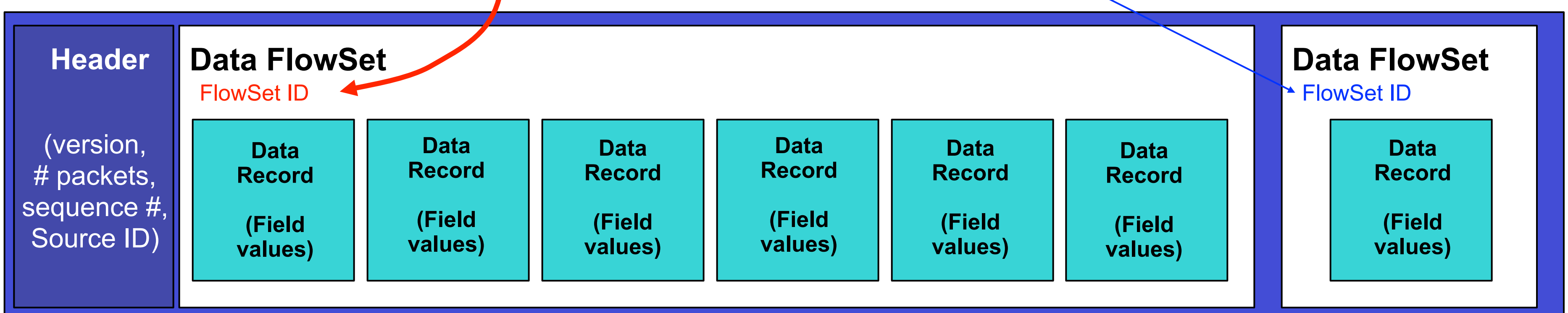
- Version 9 is an export format
- Works as a push model
- Send the template regularly (configurable)
- Independent of the underlying protocol, it is ready for any reliable protocol (e.g.,: TCP, SCTP)

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

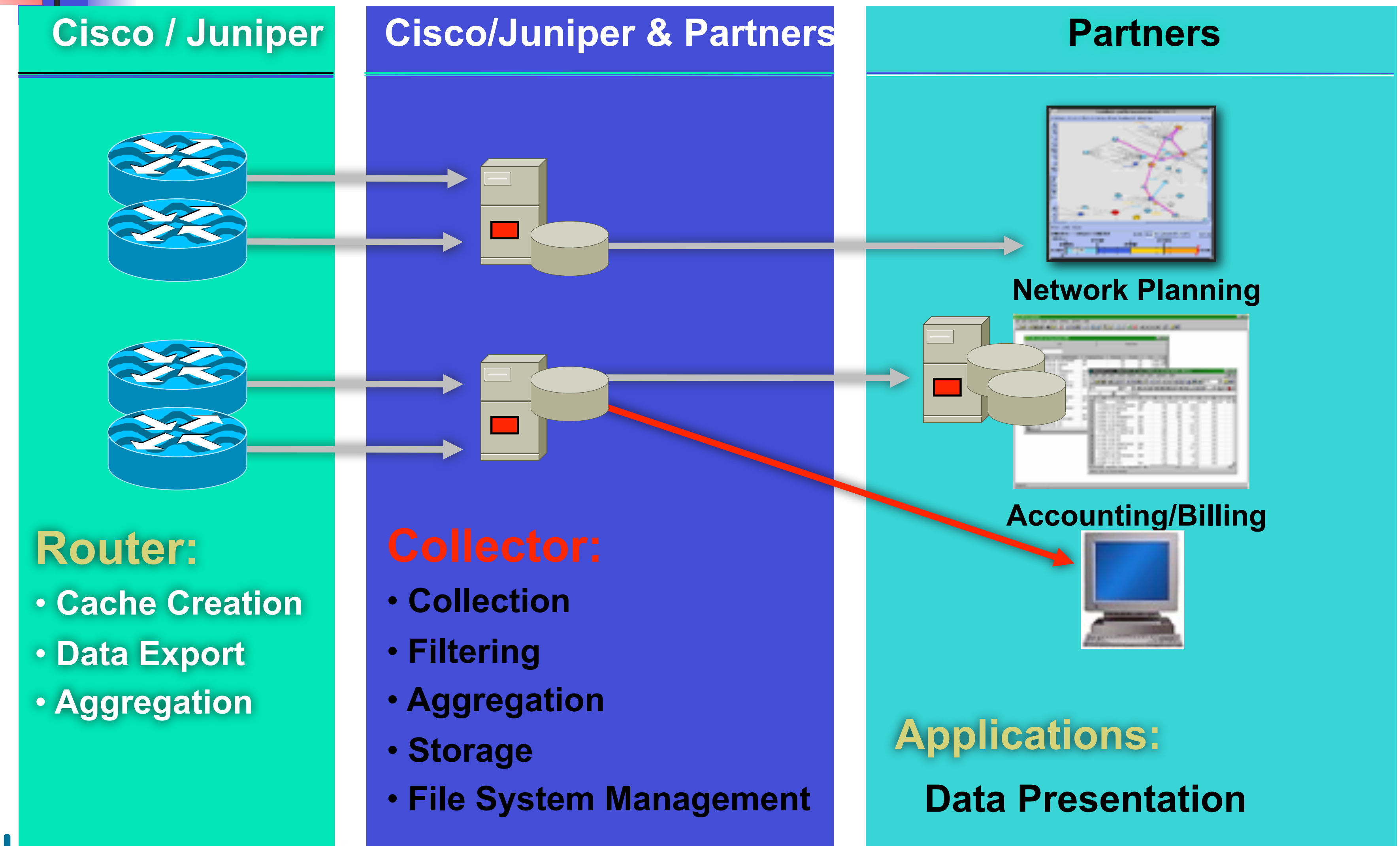
Export versions available for standard NetFlow flows

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

Export versions available for aggregated NetFlow flows

NetFlow / jflow Infrastructure



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
  4143679566 ager polls, 0 flow alloc failures
  Active flows timeout in 30 minutes
  Inactive flows timeout in 15 seconds
```

Active flows

NetFlow Timeouts – tune to avoid the churn

Cisco 7200 NetFlow Example (Cont.)

Traffic type

Protocol	Total Flows	Flows /Sec	Packets /Flow	Bytes /Pkt	Packets /Sec	Active(Sec) /Flow	Idle(Sec) /Flow
TCP-Telnet	1398292	0.3	14	156	4.6	6.0	17.2
TCP-FTP	99569986	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	

Hint: How many TCP based applications you know have 1 pkt / flow?

Cisco 7200 NetFlow Example (Cont.)

Hint: What's going on here?

```
SrcIf          SrcIPaddress    DstIf          DstIPaddress    Pr
  SrcP DstP  Pkts
Fa0/1          10.66.74.46     Fa0/0          219.103.129.162 01
  0000 0800    1
Fa0/1          10.66.115.182   Fa0/0          194.22.114.198  01
  0000 0800    1
Fa2/1          10.66.74.46     Fa0/0          61.79.227.123   01
  0000 0800    1
Fa0/1          10.66.74.46     Fa0/0          211.167.105.242 01
  0000 0800    1
Fa0/0          129.42.184.35   Null           64.104.193.198  06
  2891 0019    3
Fa2/1          10.66.115.182   Fa0/0          202.20.138.184  01
  0000 0800    1
Fa2/1          10.66.115.182   Fa0/0          63.76.237.255   01
```

Cisco Catalyst 6500 and 7600 Series Switches

```
6500>sh mls netflow ip detail
Displaying Netflow entries in Supervisor Earl
DstIP           SrcIP           Prot:SrcPort:DstPort  Src i/f:AdjPtr
--------------------------------------------------------------------
Pkts         Bytes        Age   LastSeen  Attributes
---------------------------------------------------


    QoS      Police Count Threshold    Leak     Drop Bucket  Use-Tbl Use-Enable
-----------+------------+---------+-----------+----+-------+-------+----------+
172.87.19.217   171.70.154.90   tcp :10112  :www      1023: 0
3            144          10    00:07:11   L3 - Dynamic
  0x0          0               0        0        NO   48         NO        NO
171.101.24.123  171.69.89.39    tcp :1303   :139      400 : 0
0            0            39    00:06:42   L3 - Dynamic
  0x0          0               0        0        NO   48         NO        NO
202.56.200.22   198.133.219.25  icmp:0      :0        1028: 0
26           2028         383   00:07:05   L3 - Dynamic
  0x0          0               0        0        NO   78         NO        NO
```

Review the output

Cisco Catalyst 6500 and 7600 Series Switches (Cont.)

```
6500>sh mls netflow ip dest www.cisco.com det
Displaying Netflow entries in Supervisor Earl
DstIP           SrcIP           Prot:SrcPort:DstPort  Src i/f:AdjPtr
-----------------------------------------------------------------------
Pkts         Bytes        Age   LastSeen  Attributes
---------------------------------------------------


    QoS      Police Count Threshold    Leak     Drop Bucket  Use-Tbl Use-Enable
-----------+------------+---------+-----------+----+-------+-------+----------+
198.133.219.25  66.189.188.230  icmp:0      :0        1017: 0
1            60           28    00:16:36   L3 - Dynamic
  0x0          0                0        0        NO   60         NO        NO
198.133.219.25  142.32.208.231  tcp :9415   :www      1016: 0
34           1501         32    00:16:32   L3 - Dynamic
  0x0          0                0        0        NO   40         NO        NO
198.133.219.25  65.114.202.35   tcp :4936   :www      1017: 0
24           1099         24    00:16:40   L3 - Dynamic
  0x0          0                0        0        NO   40         NO        NO
```

Review the output.

Versions

- Some releases are vendor/product specific
- What you need to know
  - Version 5
    - Widely supported
  - Version 8
    - Adds security to reporting stream (DES)
  - Version 9
    - Adds generalized formatting
    - Reduces need to upgrade tools between versions

cflowd Configuration Example

You must configure sampling for cflowd to work

```
forwarding-options {
    sampling {
        input {
            family inet {
                rate 1000;
                run-length 9;
            }
        }
        output {
            file filename sample.cfld files 20 size 1m;
            cflowd <address> {
                port <port>;
                version 5;
            }
        }
    }
}
```

```
interface FastEthernet0/0
 ip route-cache flow
interface FastEthernet0/1
ip route-cache flow

ip flow-export version 5
ip flow-export destination <ip_address> <port>
ip flow-export source FastEthernet0/0
```

Use address 169.223.142.3
Port 2x01 for Juniper
Port 2x02 for cisco
X = group number

```
interface FastEthernet0/0
 ip route-cache flow
interface FastEthernet0/1
ip route-cache flow

ip flow-export version 5
ip flow-export destination <ip_address> <port>
ip flow-export source FastEthernet0/0
```

```
forwarding-options {
     sampling {
          input {
               family inet {
                    rate 1000;
                    run-length 9;
               }
          }
          output {
               file filename sample.cfld
files 20 size 1m;
               cflowd <address> {
                    port <port>;
                    version 5;
               }
          }
     }

interfaces ge-0/1/0 {
          unit 0 {
               family inet {
                    filter {
                         input all;
                         output all;
                    }
               }
          }
}
```

Use address 169.223.142.3
Port 2x01 for Juniper
Port 2x02 for cisco
X = group number

```
firewall {
     filter all {
          term all {
               then {
                    sample;
                    accept;
               }
          }
     }
}
}
```

cflowd Output Option

- cflowd is an output option under the sampling configuration
  - Each option discussed in detail

```
forwarding-options {
    sampling {
        input {
            family inet {
                rate 1000;
                run-length 9;
            }
        }
        output {
            file filename sample.cfld files 20 size 1m;
            cflowd 10.1.86.2 {
                port 2055;
                version 5;
            }
        }
    }
}
```

cflowd Aggregate Format

Viewing the local log file on the router

```
lab@R1> show log sampled
Jan  7 18:30:44    Start time of flow: 3812598
Jan  7 18:30:44    End time of flow: 3812598
Jan  7 18:30:44    Src port: 1088
Jan  7 18:30:44    Dst port: 1241
Jan  7 18:30:44    TCP flags: 0x0
Jan  7 18:30:44    IP proto num: 6
Jan  7 18:30:44    TOS: 0x0
Jan  7 18:30:44    Src AS: 64514
Jan  7 18:30:44    Dst AS: 64513
Jan  7 18:30:44    Src netmask len: 16
Jan  7 18:30:44    Dst netmask len: 24
Jan  7 18:30:44 v5 flow entry
Jan  7 18:30:44    Src addr: 192.168.46.101
Jan  7 18:30:44    Dst addr: 172.16.3.18
Jan  7 18:30:44    Nhop addr: 10.1.84.0
Jan  7 18:30:44    Input interface: 30
Jan  7 18:30:44    Output interface: 40
Jan  7 18:30:44    Pkts in flow: 1
Jan  7 18:30:44    Bytes in flow: 46
Jan  7 18:30:44    Start time of flow: 3812603
```

```
Jan  7 18:30:44    End time of flow: 3812603
Jan  7 18:30:44    Src port: 1029
Jan  7 18:30:44    Dst port: 20
Jan  7 18:30:44    TCP flags: 0x0
Jan  7 18:30:44    IP proto num: 6
Jan  7 18:30:44    TOS: 0x0
Jan  7 18:30:44    Src AS: 64514
Jan  7 18:30:44    Dst AS: 64513
Jan  7 18:30:44    Src netmask len: 16
Jan  7 18:30:44    Dst netmask len: 24
```

Remote cflowd Server

Files created on the remote cflowd server

```
ping# ls /usr/local/arts/data/cflowd/flows
10.1.83.1.flows.0       10.1.83.1.flows.4       10.1.83.1.flows.8
10.1.83.1.flows.1       10.1.83.1.flows.5       10.1.83.1.flows.9
10.1.83.1.flows.2       10.1.83.1.flows.6
10.1.83.1.flows.3       10.1.83.1.flows.7
```

FreeBSD server running CAIDA cflowd package

Raw Flows on the cflowd Server

Viewing the raw flows on the remote cflowd server

```
ping# flowdump 10.1.83.1.flows.0
FLOW
  index:          0xc7ffff
  router:         10.1.86.1
  src IP:         192.168.46.101
  dst IP:         172.16.3.18
  input ifIndex:  30
  output ifIndex: 40
  src port:       1029
  dst port:       20
  pkts:           1
  bytes:          46
  IP nexthop:     10.1.84.0
  start time:     Mon Jan 7 21:30:12 2002
  end time:       Mon Jan 7 21:30:12 2002
  protocol:       6
  tos:            0
  src AS:         64514
  dst AS:         64513
  src masklen:    16
  dst masklen:    24
  TCP flags:      0x0
  engine type:    0
  engine id:      0
```

FreeBSD server running CAIDA cflowd package

Principal NetFlow Benefits

SERVICE PROVIDER

- Peering arrangements
- SLA VPN user reporting
- Usage-based billing
- DoS/worm detection
- Traffic engineering
- Troubleshooting

ENTERPRISE

- 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

Open Source Tools for NetFlow Analysis —The OSU Flow-Tools

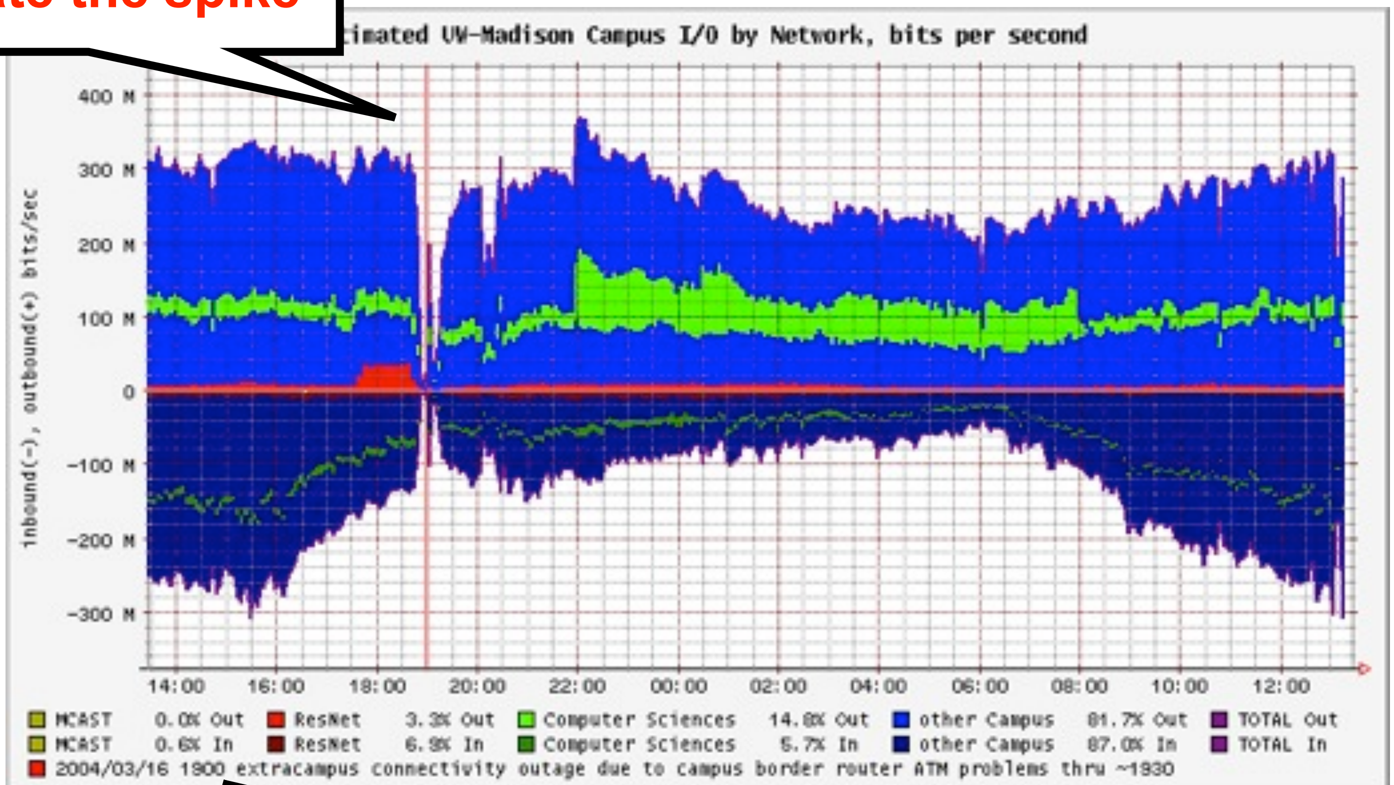
- Open source NetFlow collection and retrieval tools
- Developed and maintained by Mark Fullmer, available from http://www.splintered.net/sw/flow-tools/
- Runs on common *NIX platforms (Linux, FreeBSD, Mac OS/X, Solaris, etc.)
- Command-line tools allow for very display/sorting of specific criteria (source/dest 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!

Open Source Tools for NetFlow Analysis 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

Open Source Tools for NetFlow Analysis Visualization—FlowScan

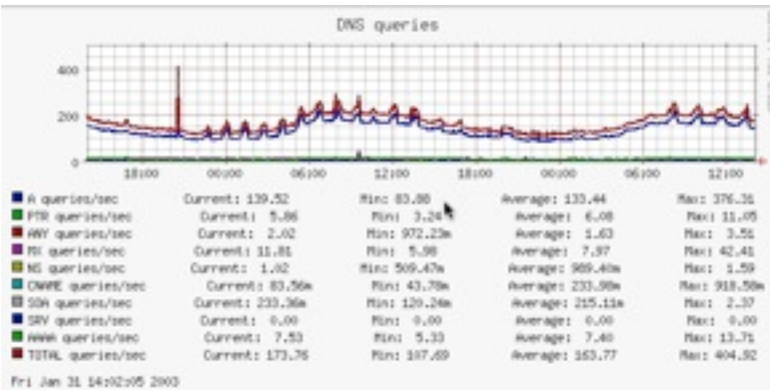
Investigate the spike

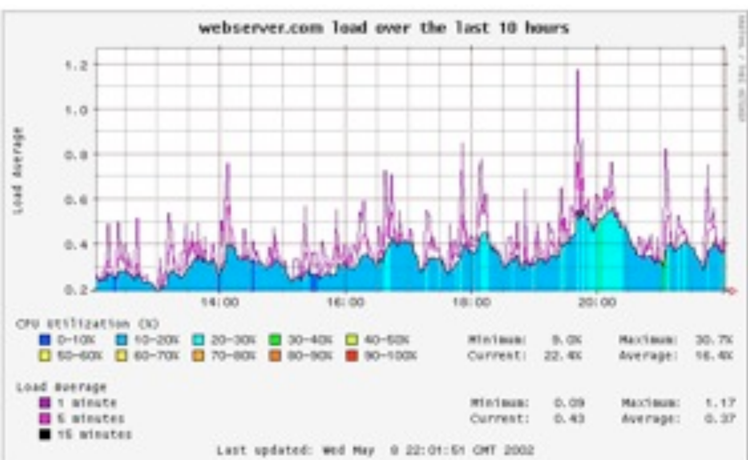


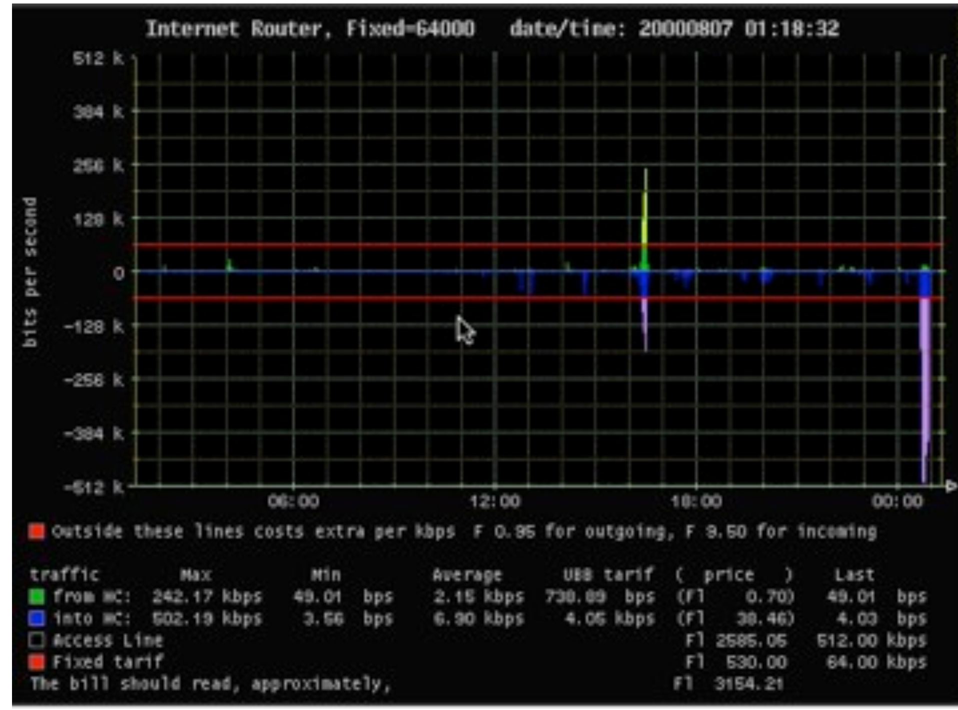
An identified cause of the outage

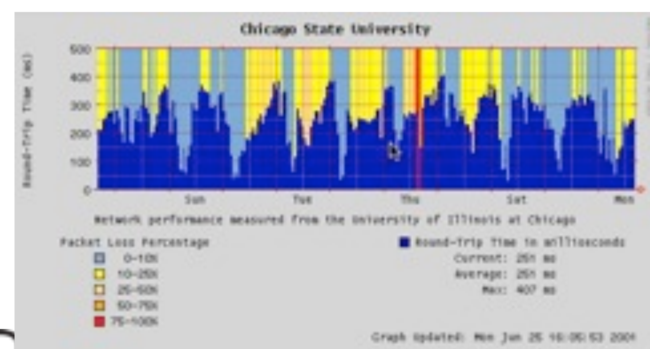
Source: University of Wisconsin

Other Visualization Techniques Using SNMP Data with RRDTool





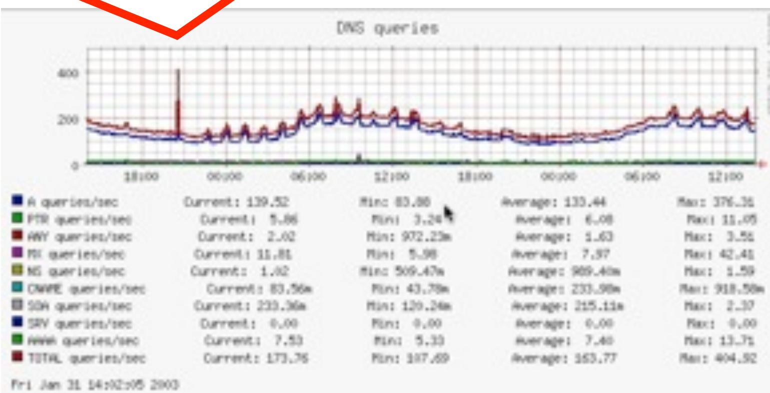


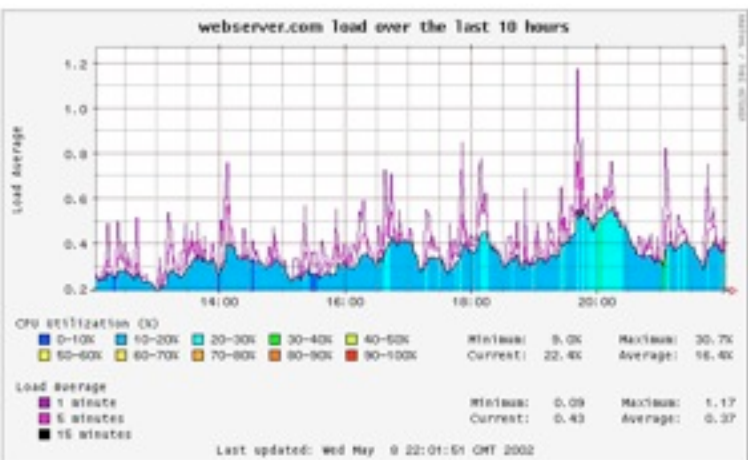


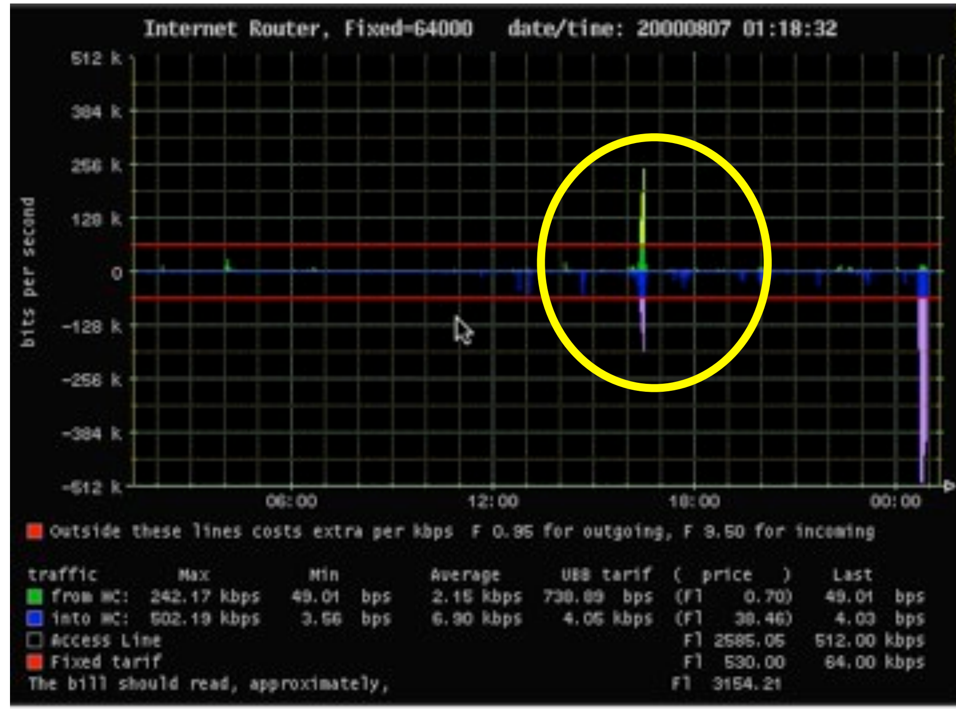
Source: http://people.ee.ethz.ch/~oetiker/webtools/rrdtool/

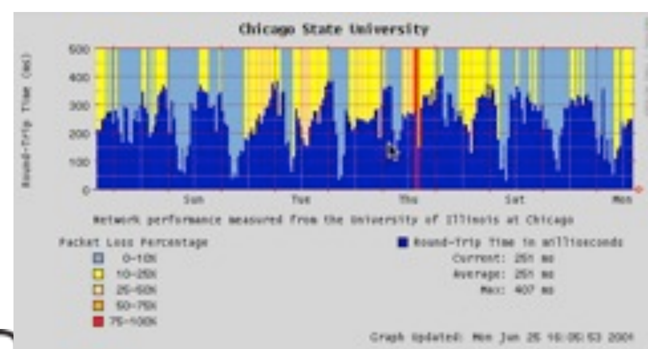
Other Visualization Techniques Using SNMP Data with RRDTool

Anomaly for DNS Queries





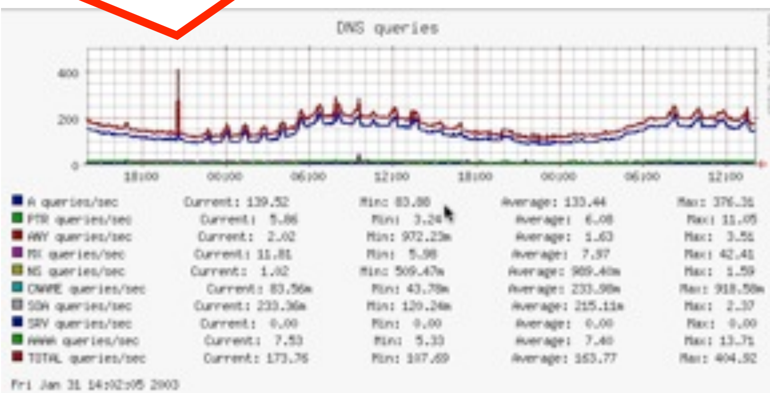


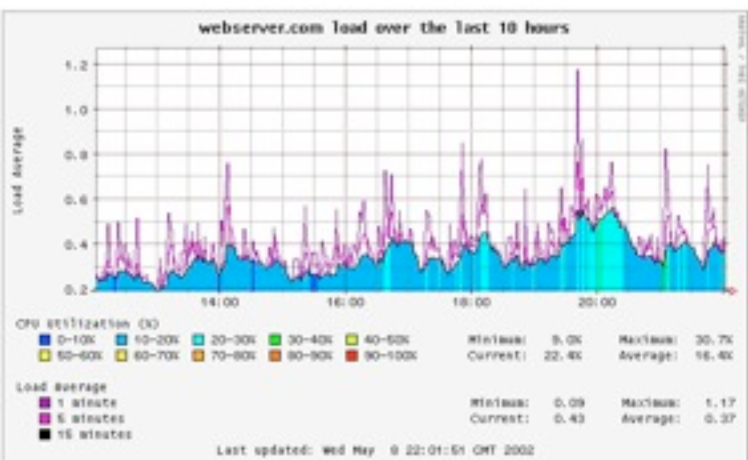


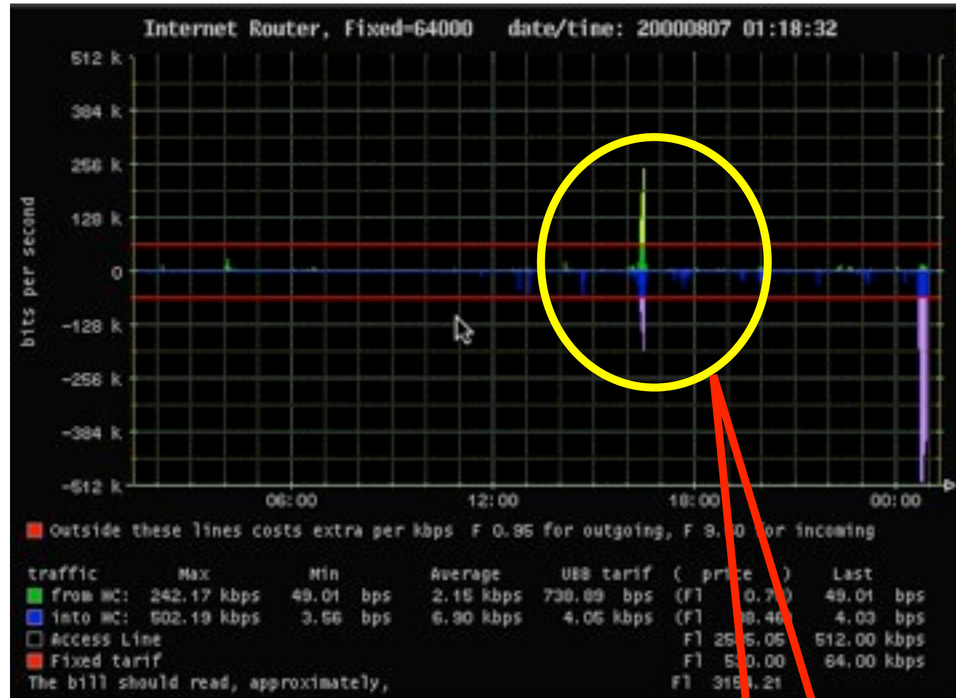
Source: http://people.ee.ethz.ch/~oetiker/webtools/rrdtool/

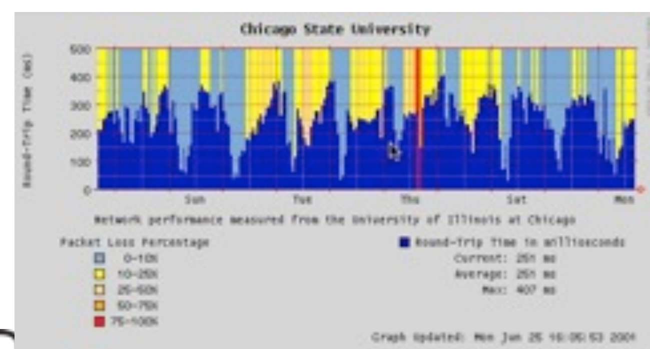
Other Visualization Techniques Using SNMP Data with RRDTool

Anomaly for DNS Queries







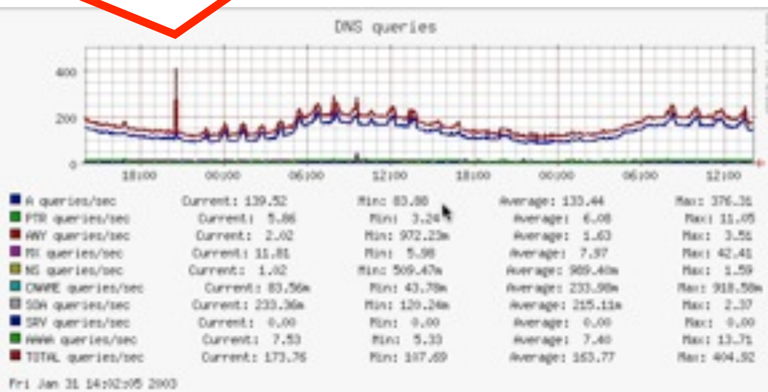


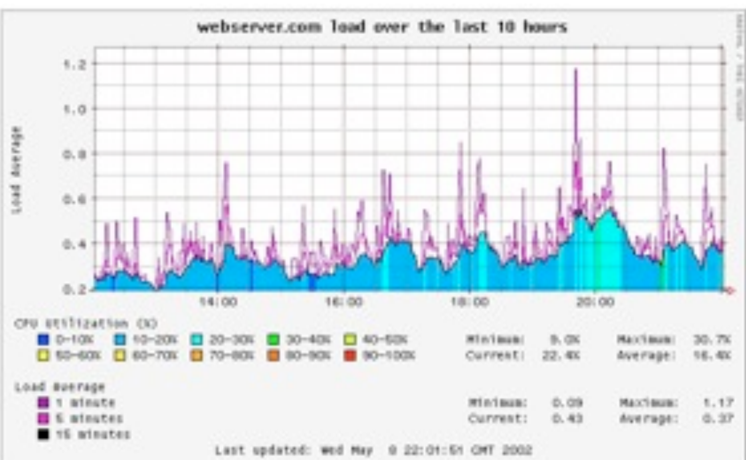
Thru'put Spike

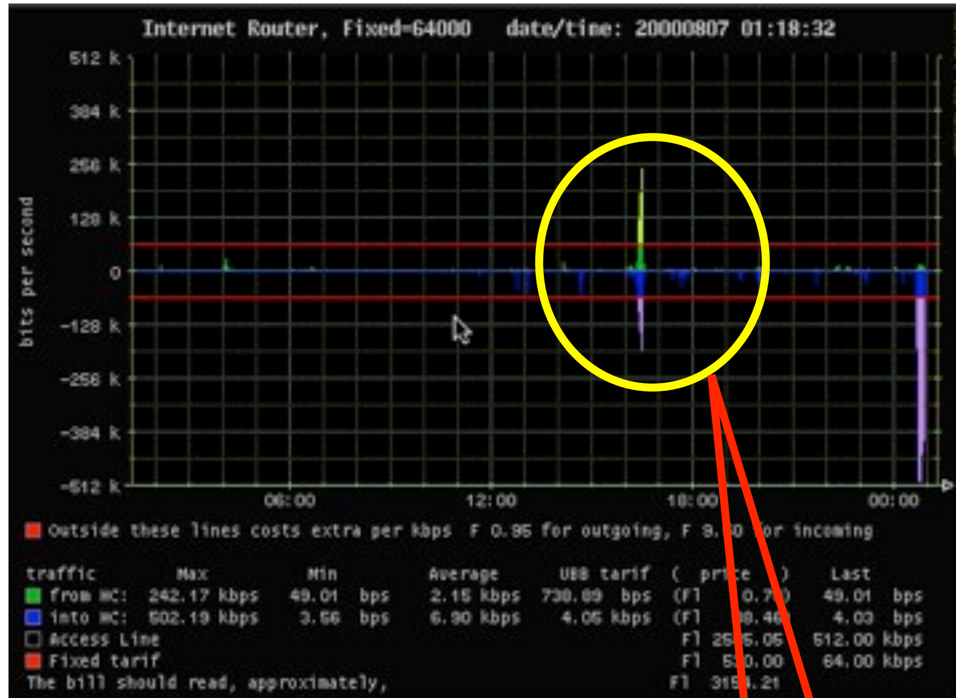
Source: http://people.ee.ethz.ch/~oetiker/webtools/rrdtool/

Other Visualization Techniques Using SNMP Data with RRDTool

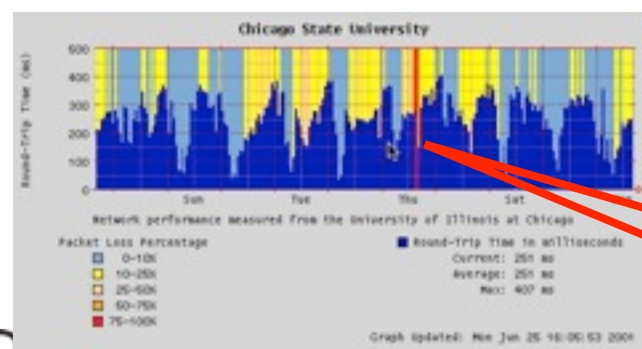
Anomaly for DNS Queries







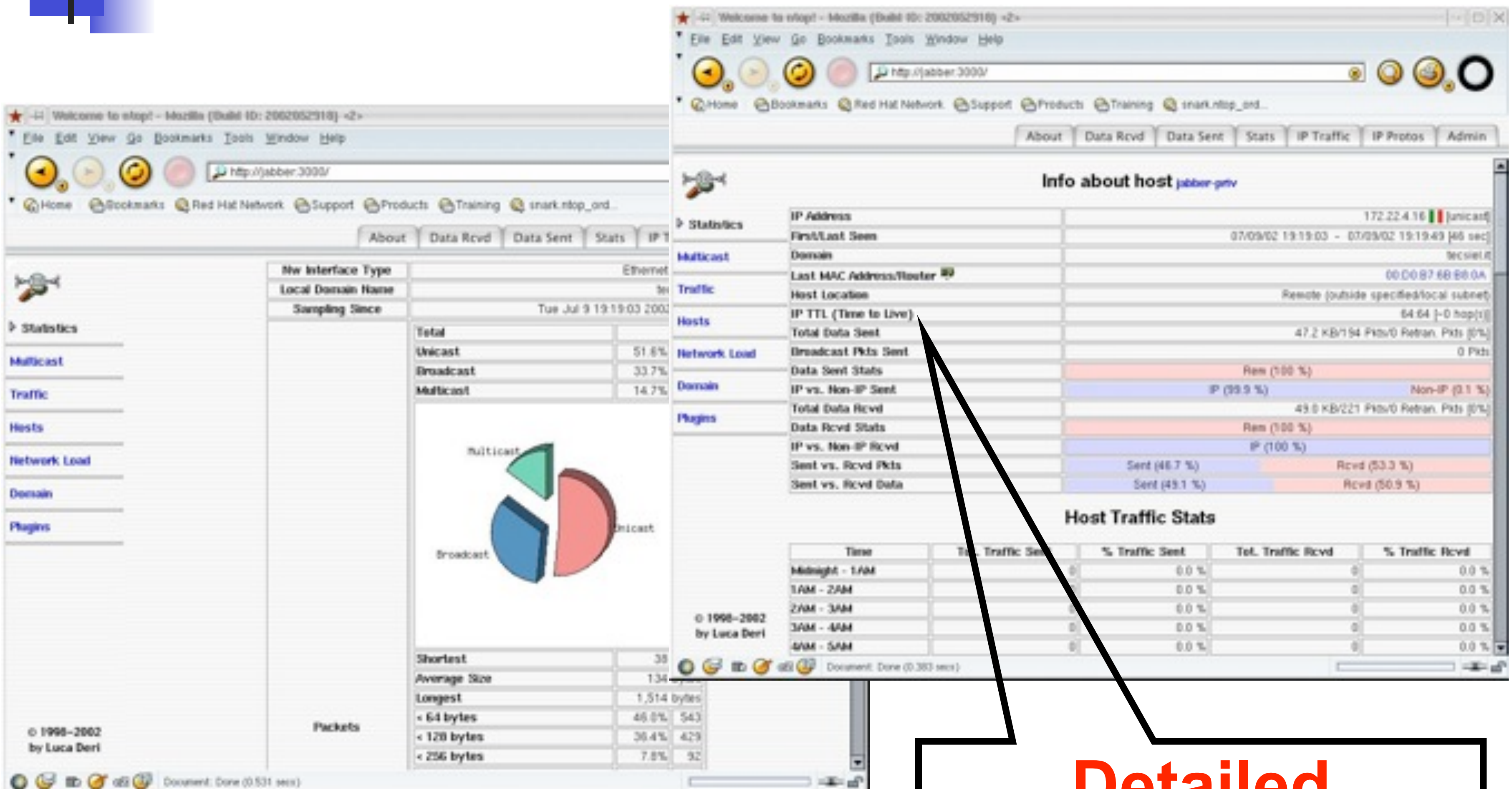
Thru'put Spike



RTT Spike

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

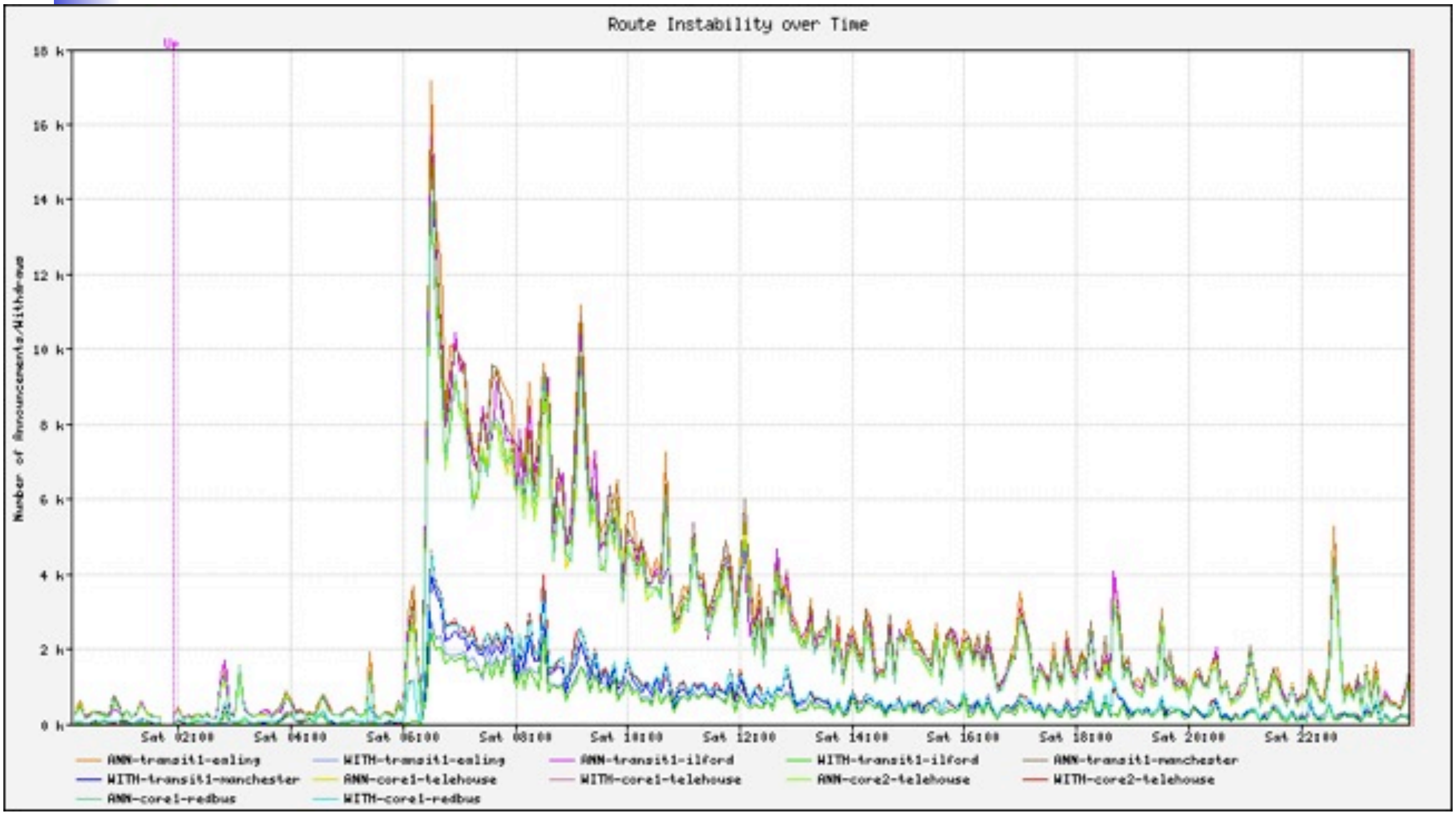
Displaying RMON—ntop Examples



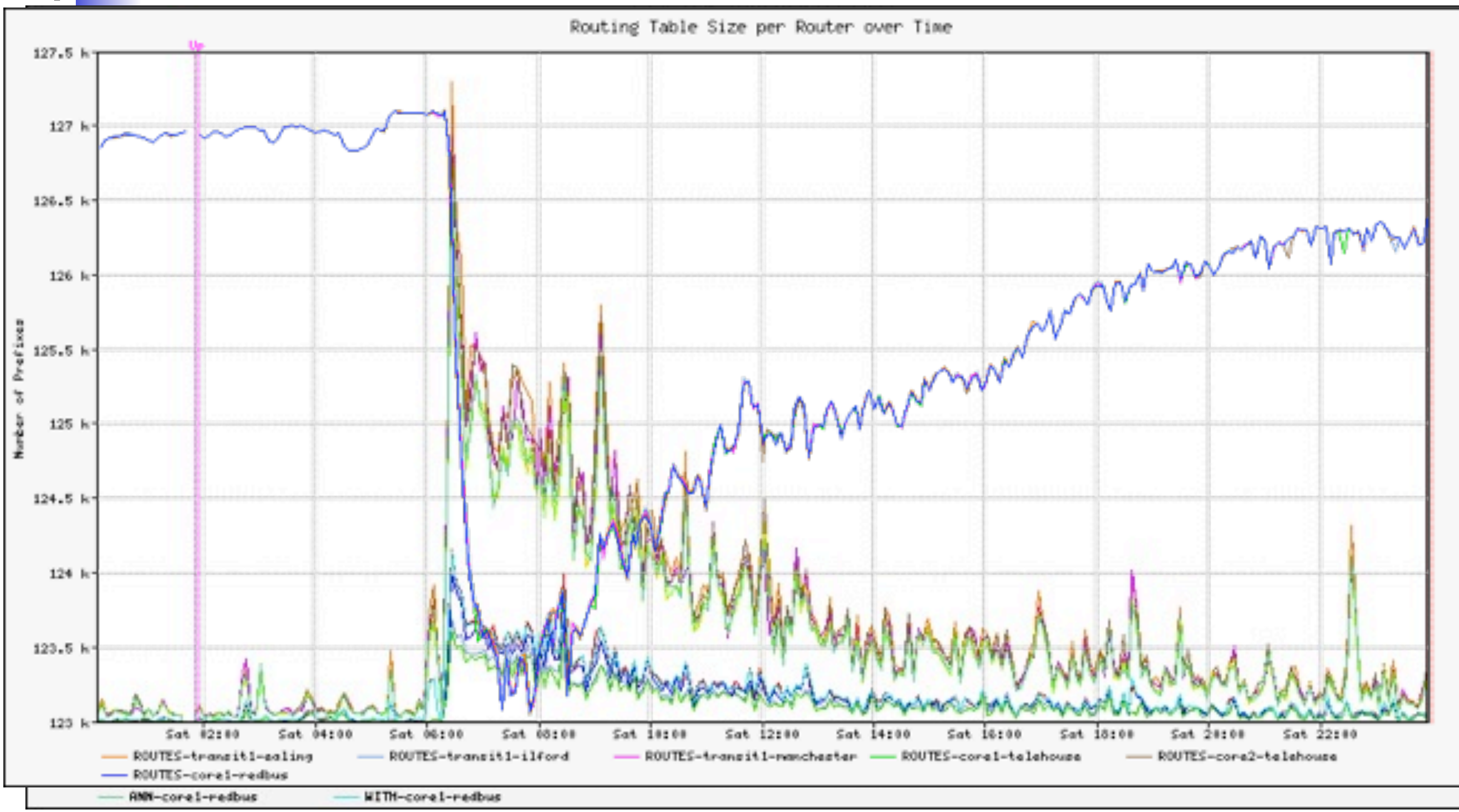
Detailed Analysis i.e. TTL

Source: http://www.ntop.org

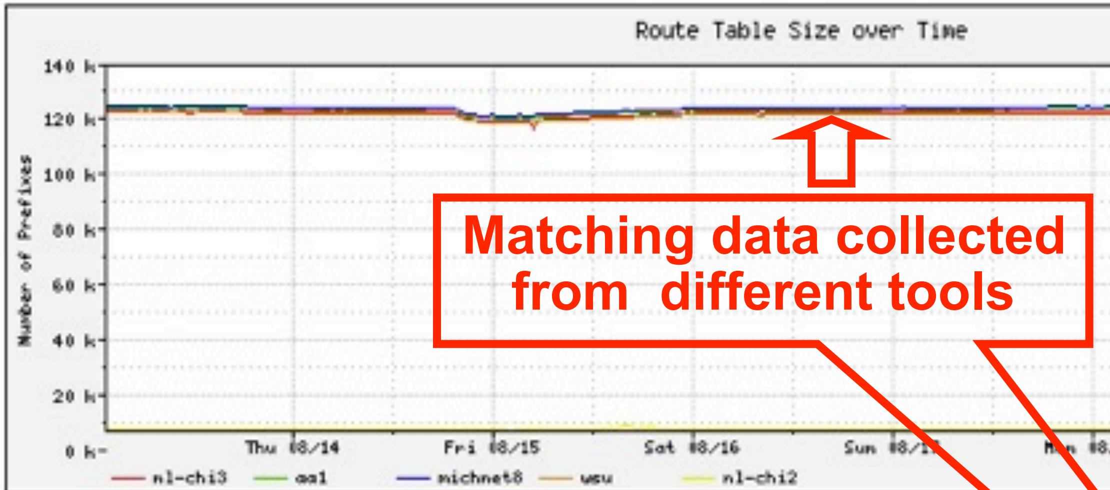
BGP Example—SQL Slammer



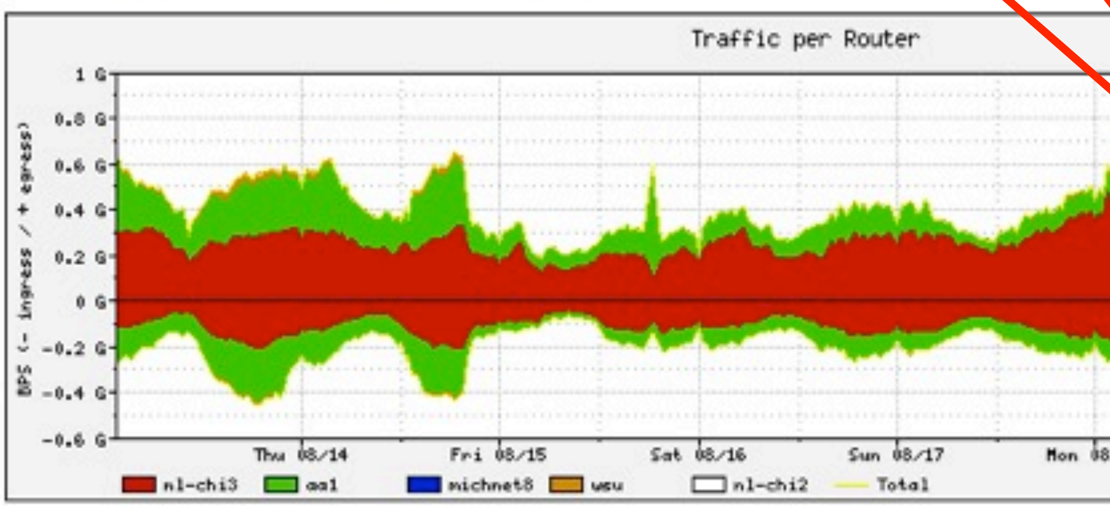
BGP Example—SQL Slammer



Correlating NetFlow and Routing Data



Matching data collected from different tools




```
tcsh — tcsh
danny@rambler% cat prefixes

Prefix                  Daily        Daily
Length   *Current       Max          Average

/24      65,900         68,497       67,259
/23       9,904         10,157       10,027
/22       9,053          9,211        9,110
/21       6,035          6,106        6,045
/20       8,405          8,560        8,487
/19       8,175          8,221        8,161
/18       3,007          3,031        3,005
/17       1,693          1,705        1,690
/16       7,293          7,396        7,326
/15         473            473          469
/14         263            263          262
/13          98             98           97
/12          55             55           54
/11          12             12           11
/10           6              6            5
/9            4              4            3
/8           19             19           18

Current_Total:    120,475
Max_Total:        123,814
Average_Total:    122,029

Current v. Average:     98.73% (1554 prefixes)

* Current Based on my Snapshot @9P MDT 8.14.2003
[~]
danny@rambler%
```

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
- Logging of ACLs is generally contraindicated due to CPU overhead—NetFlow provides more info, 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 info based upon device type, function, other criteria
- Syslog-ng from http://www.balabit.com/products/syslog_ng/ adds a lot of useful functionality—HOW-TO located at http://www.campin.net/newlogcheck.html

Local Log Files

- Local log files are useful for:
  - Detecting problems on the router
  - Monitoring the system usage by friendly users
  - Monitoring normal events
- Local log files are not useful for:
  - Monitoring the activity of attackers who have compromised your system

A good attack will erase the evidence of their activity from local log files!

Remote Log Files

- Logging to a remote host has several advantages:
  - Initial attacker activity is available in the log
    - Remote logging is difficult to stop prior to the compromise
    - Attacker can stop remote logging once the system is compromised
    - The lack of remote logging can be an indication of a problem
  - Remote logs from multiple systems can be consolidated

What is Syslog?

- Operating systems and applications generate a multitude of log messages about a variety of things
  - Syslog was developed as a generic logging server to accept, categorize, and record log messages
  - As systems became more complex, a method was needed to forward log messages to a remote syslog server and consolidate messages from multiple hosts
  - BSD Syslog Protocol
    - Outlined in RFC 3164
    - Specifies the format and content of remote syslog messages

Syslog Facilities (1 of 2)

- Each message has a facility used to categorize the type of message generated
- The router specifies the facility to which each message belongs

Facility	Description
Any	Any facility
Authorization	Any authorization attempt
Change-log	Any change to the configuration
Conflict-log	Messages generated when configuration conflicts with the hardware
Cron	Cron daemon
Daemon	Various system daemons

Syslog Facilities (2 of 2)

Facility	Description
Firewall	Firewall filtering subsystem
Interactive-commands	Commands executed in the CLI
Kernel	Messages generated by the JUNOS software kernel
PFE	Messages generated by the PFE
User	Messages from user processes
Local0 – Local7	Local-use facilities

Syslog Severity

- Each message has a severity used to prioritize its importance
  - Setting a facility and severity level causes the router to log all messages for that severity at the specified level and above
    - For example, logging at the `critical` level also causes `alert` and `emergency` messages to be logged

emergency alert critical error warning notice info debug

More severe

Syslog Packet Format

- Format and content of messages defined in RFC 3164
  - Messages sent on UDP port 514
  - No minimum size
  - Maximum size is 1024 bytes
- Messages consist of three text strings
  - PRI (Priority)
  - HEADER
  - MSG (Message)

PRI	**HEADER**
MSG	

Overriding the Remote Facility

- By default, syslog messages are sent with their normal BSD-specified facility and various local facilities
  - You can override the message facility

```
[edit system]
lab@R1# show
syslog {
    host 10.1.10.2 {
        authorization info;
        change-log info;
        interactive-commands info;
        facility-override local7;
        log-prefix Security;
    }
}
```

Security Cautions

- Caution:
  - Syslog messages can contain sensitive information in cleartext
    - User authentication messages when logging the authorization facility
    - Passwords entered into the configuration when logging the interactive-commands facility
  - Consider sending syslog messages only on the out-of-band management network
  - Compromise of the remote syslog server might give an attacker enough information to compromise the router!

Good things to log

- All login attempts
  - Successful or not
- All commands typed
  - So you know who did what and when
  - Helps Identifying "training issues" as well ☺
- Availability issues
  - Interface status change (not dialup)
  - More critical for core routers
  - BGP peering changes
  - OSPF neighbor changes?

NTP

Benefits of Deploying NTP

- Very valuable on a global network with network elements in different time zones
- Easy to correlate data from a global or a sizable network with a consistent time stamp
- NTP based timestamp allows to trace security events for chronological forensic work
- Any compromise or alteration is easy to detect as network elements would go out of sync with the main 'clock'
- Did you there is an NTP MIB? Some think that we may be able to use "NTP Jitter" to watch what is happening in the network.

Local System Time

- In a security situation, you must have a consistent concept of time across the network (it does not have to be the correct time, just consistent)
- Choose UTC/GMT or Head office Time Zone
- NTP was developed to synchronize large numbers of network devices to a consistent, accurate time reference
- Local and remote log files are stamped with the local system time
  - Event correlation is easier if all devices are synchronized
  - Law enforcement officials might need copies of these logs

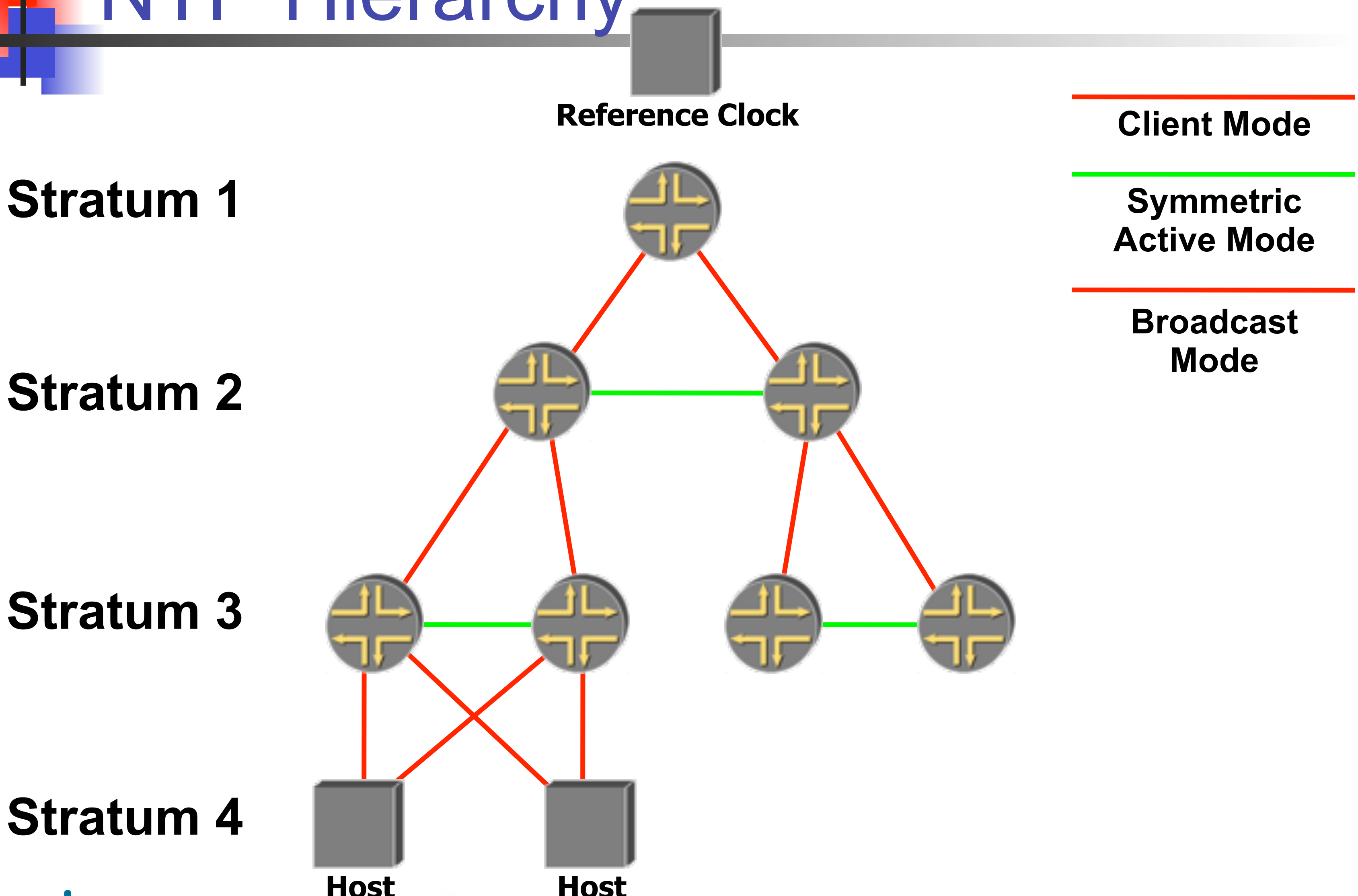
Network Time Protocol

- NTPv3: Network Time Protocol (Version 3) Specification, Implementation and Analysis (RFC 1305—March 1992)
  - Defines a protocol to keep accurate, synchronized time between network devices
    - Uses UDP port 123
  - Additional features incorporated in NTPv4
    - DES encryption
    - Not an IETF standard, but widely supported
    - Backwards compatible with NTP v3

Three NTP Modes

- Three modes:
  - Client mode
    - Client synchronizes local time one way to remote server
  - Symmetric active mode
    - Equal peer systems synchronize each other’s local time
  - Broadcast mode
    - Server sends periodic broadcast/multicast messages on broadcast-capable media
    - Clients receives broadcast/multicast messages and synchronize local time

NTP Hierarchy



Stratum

Stratum	Min accuracy
1	1.0×10^{-11}
2	1.6×10^{-8}
3	4.6×10^{-6}

NTP Security

- NTP security
  - NTP relies on the number of connected hosts to:
    - Receive accurate time information
    - Isolate participants whose clock is incorrect
  - NTP supports MD5 and DES authentication
  - NTP without authentication on the public network is subject to spoofing
    - Create the appropriate filters to block incoming unsolicited information
    - Consider using the management network for NTP traffic

NTP Boot Server

- NTP particulars:
  - NTP will not synchronize with a peer whose time is very different
    - Tiny offsets are adjusted normally
    - Small offsets are slewed (adjusted slowly)
    - Larger offsets are stepped (set anew)
    - Huge offsets are rejected outright
  - To synchronize the initial time:
    - Use an NTP boot server
    - When the router is booted a request is issued to the boot server to get the initial reference time

```
[edit system]
lab@R1# show
ntp {
    boot-server 10.1.10.2;
```

Client Configuration

```
[edit system]
lab@R1# show
ntp {
    boot-server 10.1.10.2;
    server 10.1.10.2 version 3
  prefer;
    server 10.1.9.2;
}
```

Symmetric Active Mode Configuration

```
[edit system]
lab@R1# show
ntp {
    boot-server 10.1.10.2;
    peer 10.1.10.2 version 3 prefer;
    peer 10.1.9.2;
}
```

Broadcast Mode Configuration

```
[edit system]
lab@R1# show
ntp {
    boot-server 10.1.10.2;
    server 10.1.10.2 version 3
  prefer;
    peer 10.1.9.2;
    broadcast 224.0.1.1;
    broadcast 10.1.2.255 version 3;
```

Broadcast Client Configuration

```
[edit system]
lab@R1# show
ntp {
    boot-server 10.1.10.2;
    broadcast-client
}

[edit system]
lab@R1# show
ntp {
    boot-server 10.1.10.2;
    multicast-client
}
```

Authentication

- Authentication of time synchronization
  - All NTP modes can use authenticated connections
    - Prevents spoofing
  - Supports two types of encrypted/hashed authentication algorithms
    - DES
    - MD5
  - Easy to configure
    - No real reason not to use authentication

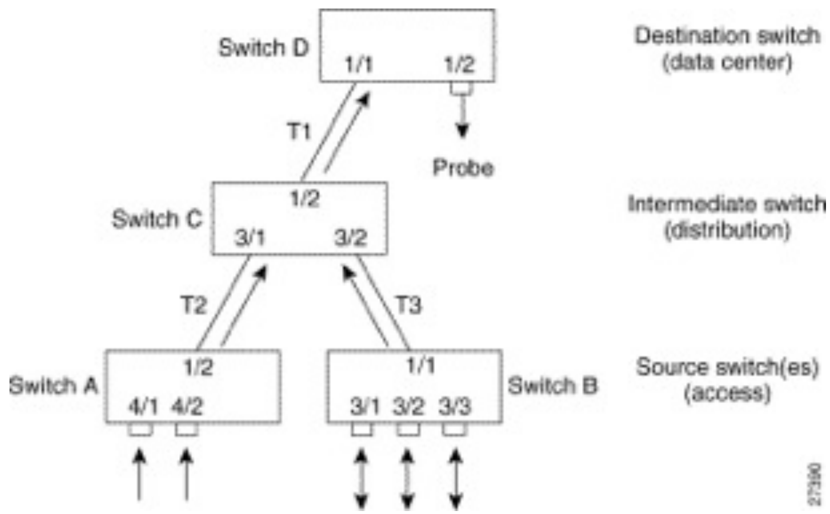
Utilizing Packet Capture

- SPAN/RSPAN (6500/7600, 4K, 2900,), copy/capture VACLs (6500/7600), IP Traffic Export (software-based routers) are all used to get packets to analysis systems
- SPAN/RSPAN and copy/capture VACLs do not have measurable performance impact; IP Traffic Export can delay processing of traffic outbound from the router, based upon the volume of traffic to be replicated
- A *NIX box running tcpdump is a common method of capturing packets, with analysis performed offline using additional open-source tools such as Ethereal
- The Cisco NAM-2 captures packets via SPAN/RSPAN or copy/capture VACLs on the 6500/7600; it can perform basic on-board analysis, but captures are typically saved and downloaded for use in Ethereal, Network General Sniffer, etc.
- Packet capture is generally undertaken after a macro-level

Utilizing Packet Capture (cont.)

- Packet capture should take place at key points in the topology such as distribution gateways, IDC switch meshes, desktop access switch meshes, and in some cases, the core
- It is important to be as specific as possible when capturing packets; at high rates of speed, the amount of information can be overwhelming
- There's lots of garbage out there - 'weird' packets are often perfectly explicable, in context
- It's extremely important to ensure that traffic is captured bidirectionally - or, if this isn't possible, the observer must know about the unidirectionality of the capture and take it into account when analyzing the captured traffic
- Conversely, it's important to avoid capturing duplicate

Packet Capture Example - CatOS RSPAN

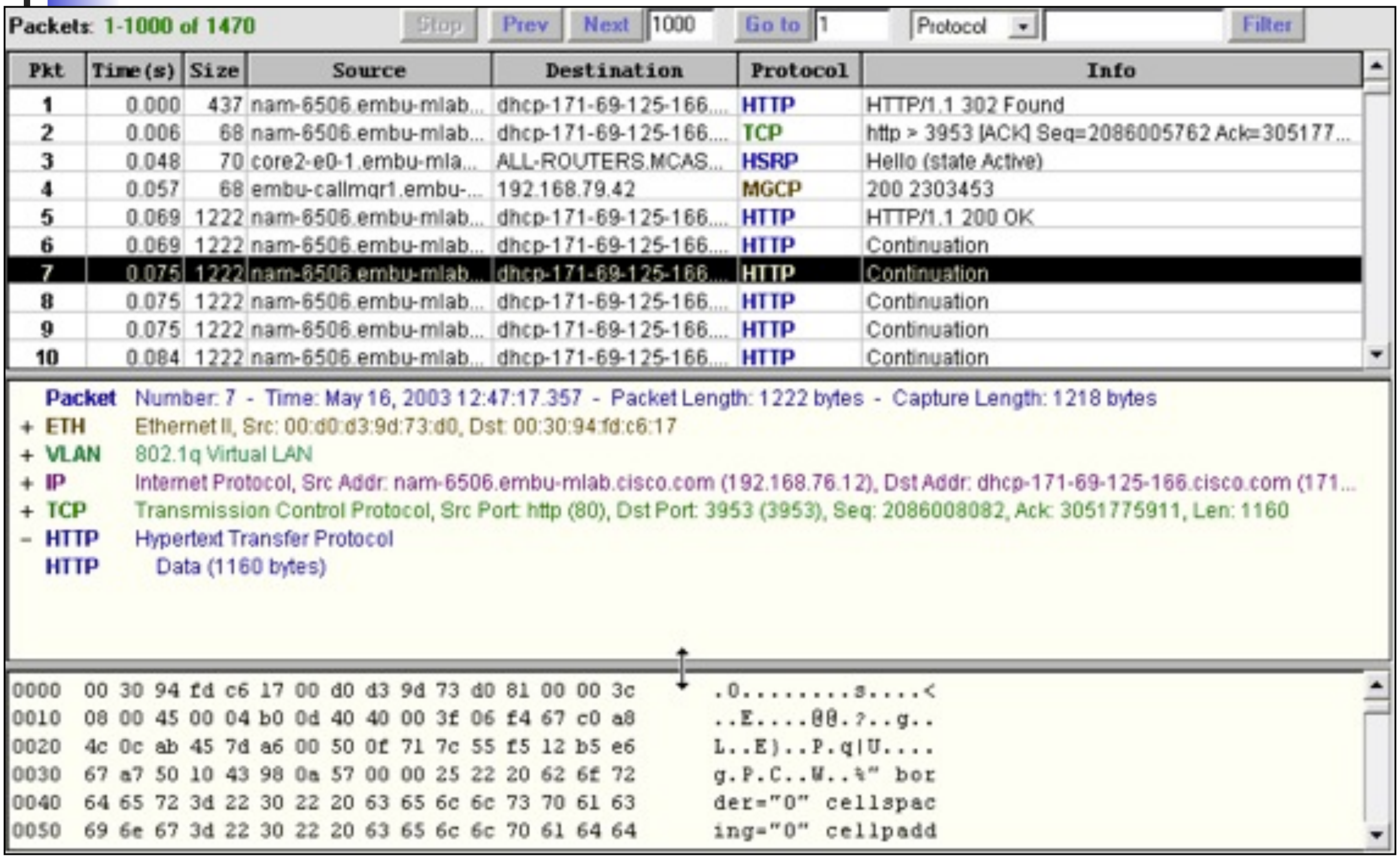


Switch	Ports	RSPAN VLAN	Direction	RSPAN CLI Commands
A (source)	4/1, 4/2	901	Ingress	**set rspan source 4/1-2 901 rx**
B (source)	3/1, 3/2, 3/3	901	Bidirectional	**set rspan source 3/1-3 901**
C (intermediate)	-	901	-	No RSPAN CLI command needed
D (destination)	1/2	901	-	**set rspan destination 1/2 901**

Packet Capture Example - tcpdump

```
tcpdump -lllvvvnxxXX -s 1500 -i en1
tcpdump: listening on en1, link-type EN10MB (Ethernet), capture size 1500 bytes
..
07:10:25.740130 IP (tos 0x0, ttl  64, id 15460, offset 0, flags [none], length: 59) 10.25.7.122.58607
    > 172.17.168.183.53: [udp sum ok]  15197+ A? delta.mac.com. (31)
        0x0000:  0005 31a0 3414 000d 93f0 c5bc 0800 4500  ..1.4.........E.
        0x0010:  003b 3c64 0000 4011 d8bd 0a19 077a ab46  .;<d..@......z.F
        0x0020:  a8b7 e4ef 0035 0027 bfb9 3b5d 0100 0001  .....5.'..;]....
        0x0030:  0000 0000 0000 0564 656c 7461 036d 6163  .......delta.mac
        0x0040:  0363 6f6d 0000 0100 01                   .com.....
07:10:25.829524 IP (tos 0x0, ttl  56, id 14524, offset 0, flags [DF], length: 256) 172.17.168.183.53
    > 10.25.7.122.58607: [udp sum ok]  15197 q: A? delta.mac.com. 2/4/4 delta.mac.com. CNAME
    idisk.mac.com., idisk.mac.com. A 17.250.248.77 ns: mac.com. NS nserver4.apple.com., mac.com. NS
    nserver.apple.com., mac.com. NS nserver2.apple.com., mac.com. NS nserver3.apple.com. ar:
    nserver.apple.com. A 17.254.0.50, nserver2.apple.com. A 17.254.0.59, nserver3.apple.com. A
    17.112.144.50, nserver4.apple.com. A 17.112.144.59 (228)
        0x0000:  000d 93f0 c5bc 0005 31a0 3414 0800 4500  ........1.4...E.
        0x0010:  0100 38bc 4000 3811 a3a0 ab46 a8b7 0a19  ..8.@.8....F....
        0x0020:  077a 0035 e4ef 00ec c78e 3b5d 8180 0001  .z.5......;]....
        0x0030:  0002 0004 0004 0564 656c 7461 036d 6163  .......delta.mac
        0x0040:  0363 6f6d 0000 0100 01c0 0c00 0500 0100  .com............
        0x0050:  0006 ea00 0805 6964 6973 6bc0 12c0 2b00  ......idisk...+.
        0x0060:  0100 0100 000d da00 0411 faf8 4dc0 1200  ............M...
        0x0070:  0200 0100 0222 ab00 1108 6e73 6572 7665  .....".....nserve
        0x0080:  7234 0561 7070 6c65 c016 c012 0002 0001  r4.apple........
        0x0090:  0002 22ab 000a 076e 7365 7276 6572 c058  ..".....nserver.X
        0x00a0:  c012 0002 0001 0002 22ab 000b 086e 7365  ........"....nse
```

Packet Capture Example - Ethereal

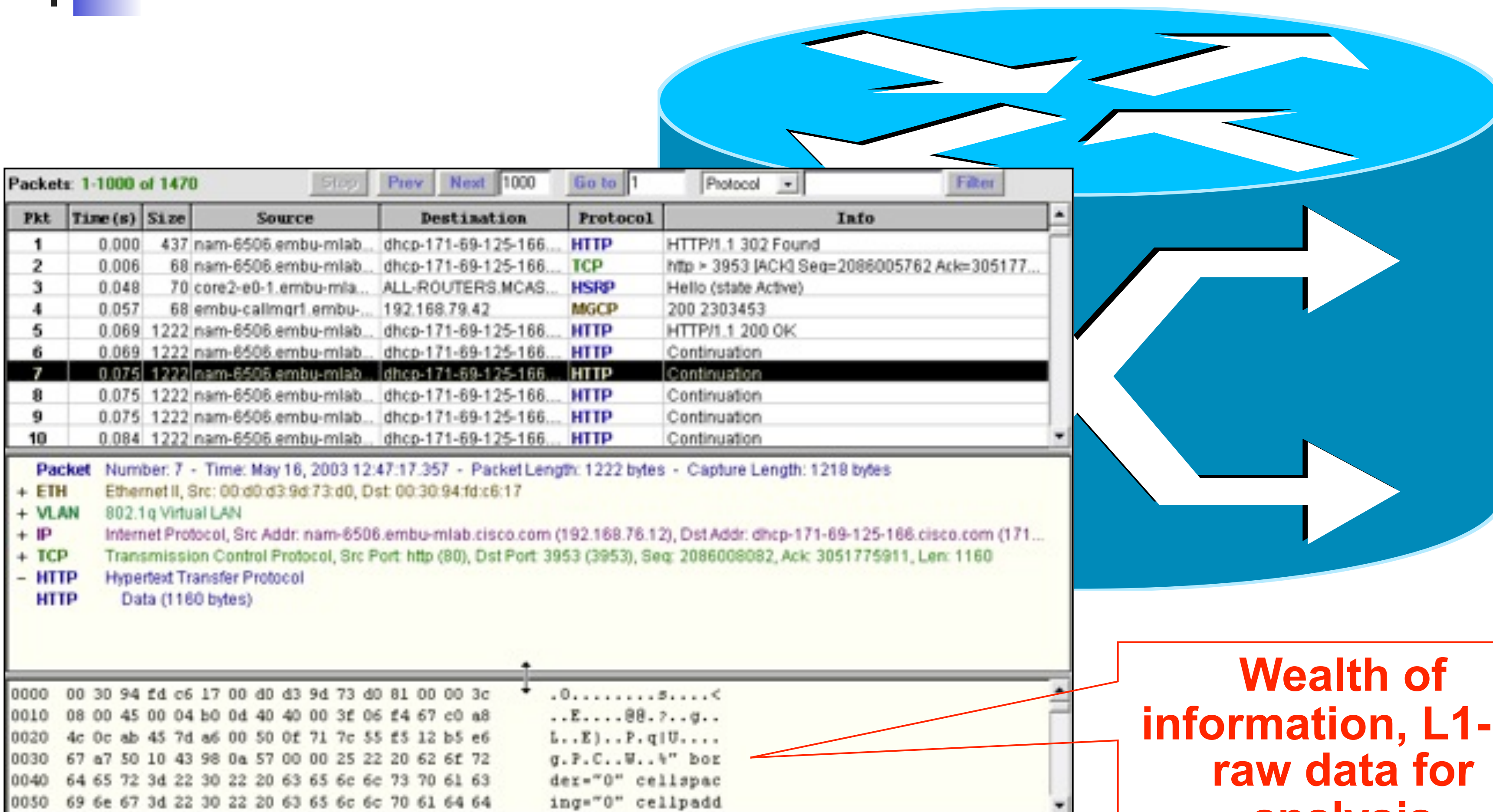


Source: http://www.ethereal.com

References

- 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]

Packet Capture Examples



Wealth of information, L1-L7 raw data for analysis

Source: http://www.ethereal.com, Cisco Systems, Inc.

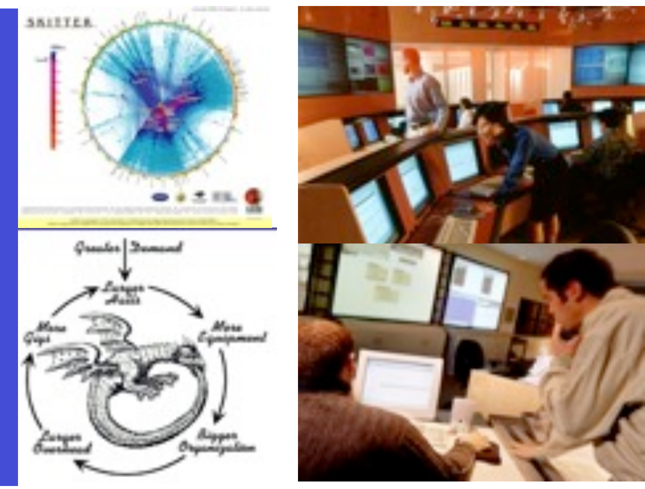
Tell Me Where to Start From?

1. NetFlow or jflow enablement on the network elements
2. NetFlow or jflow data correlation and analysis
3. SNMP / RMON [SNMP more prevalent]
   1. CPU / Memory util
   2. Link usage and display with MRTG
4. SysLog collection and analysis
5. Monitoring to Routing, DNS queries, etc. [BGP, DNS]
6. Local and remote packet capture facility [Most have it today with sniffer, ethereal]

Homework from Total Visibility

- Define telemetry strategy—ASAP
  - Local and remote
- Need to start deployment today where the most bang for the buck is offered. However, the end goal is to achieve the holistic view
- Telemetry: Deploy, Understand and Practice
  - For any security event – Proactive Telemetryor telemetry during the incident, if 'SECOPS' trained then they can use it with familiarity of 'back of their hand'
- Telemetry builds foundation to be successful with all the other 5 of 6 steps methodology

MPLS / L3VPN Security



Before we start…

- Mainly of interest to providers/ISP/Carriers
  - Some interest in enterprise
- To support MPLS in your network you **MUST** have
  - Fully working IP network. If it's broken MPLS won't fix it.
  - Hardware and Software support. Depends on vendors
    - Juniper
      - All our routers (M-Series, T-Series, J-Series, E-Series)
    - Cisco
      - CEF support

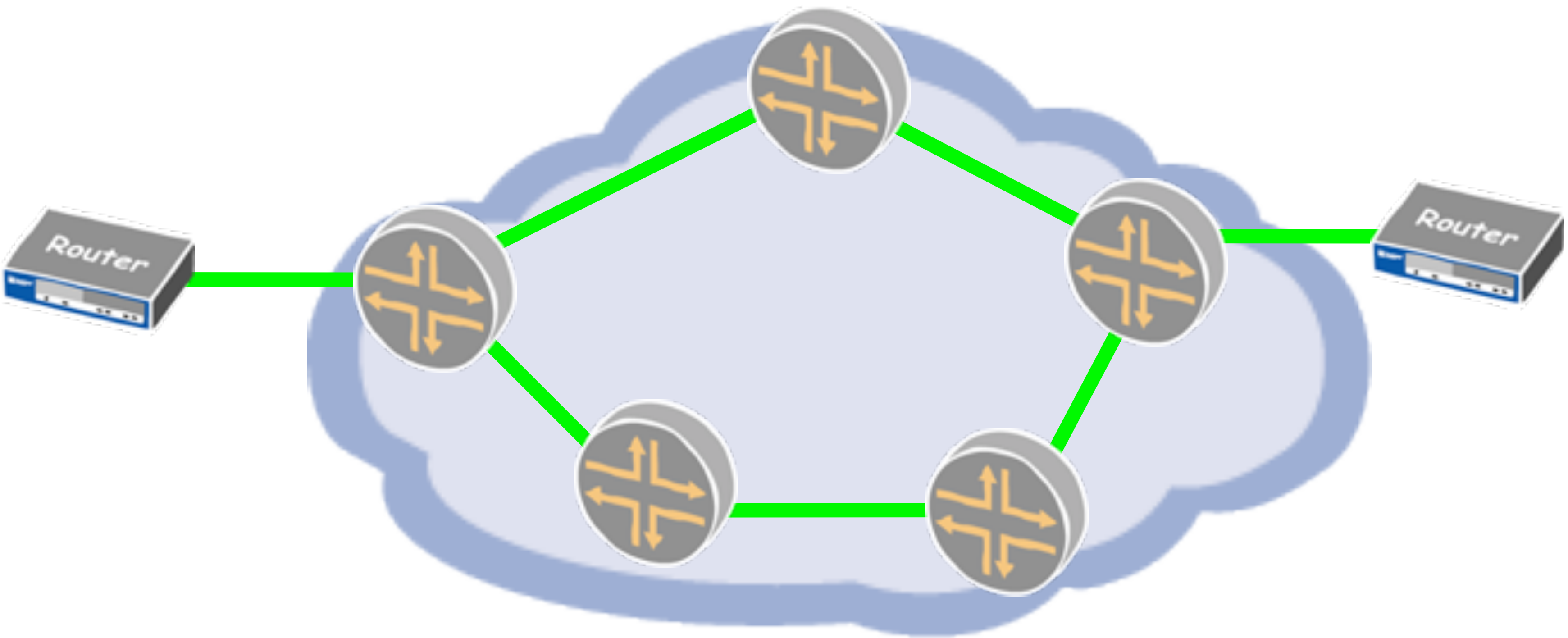
Things I want you to know

- MPLS is a tool to solve problems
  - Not everyone has the same problems or pain
- In other words reason to deploy (choose 1+)
  - Traffic Engineering
  - Traffic Protection
  - Provider provisioned VPN's
    - Layer 3 and/or Layer 2
- Or in other words
  - Save money
  - Make money

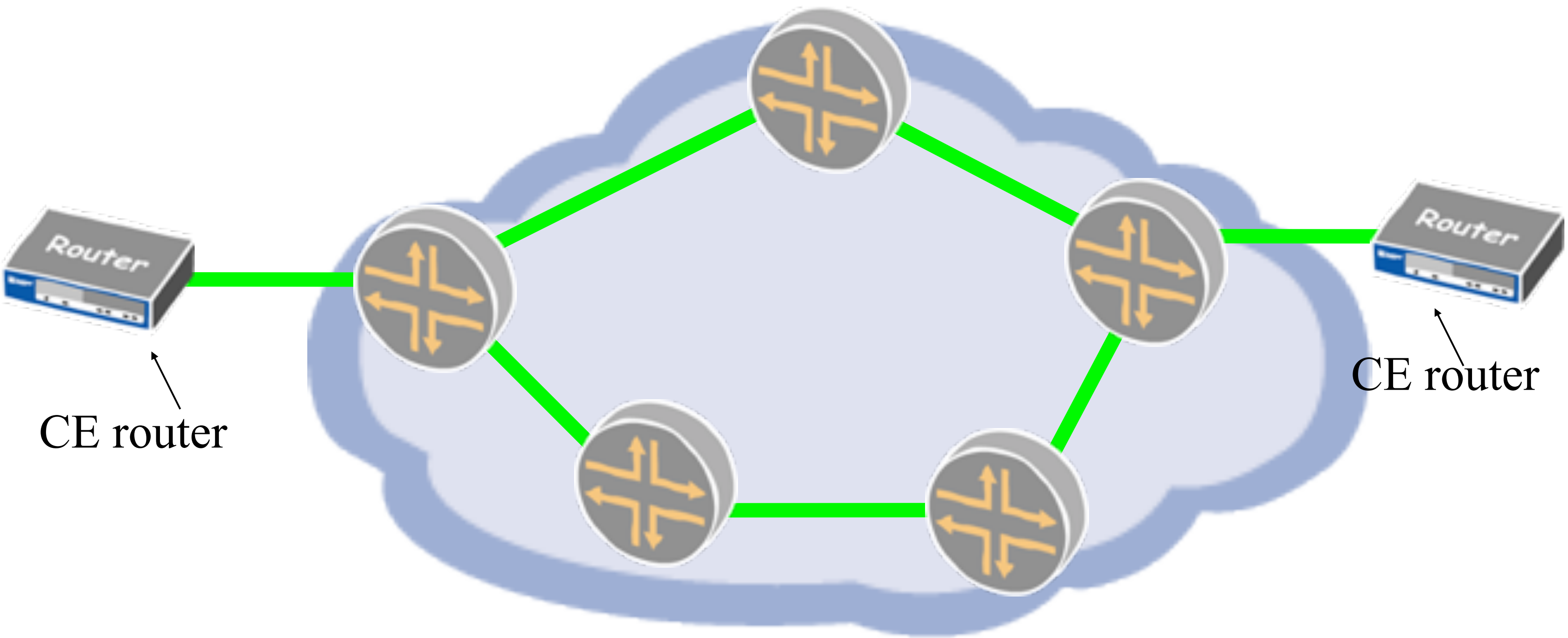
What is MPLS?

- Multiprotocol Label Switching
- Connection Orientated Virtual Circuits over IP implemented with label switching
- Grew out of
  - Cisco's Tag switching
  - Ipsilon (Nokia) IP switching
  - IBM ARIS
  - 3Com's FAST IP
- Expanding area's of application
  - Cost savings
  - New services
- Promise of Multiprotocol Unification (Core NOT edge)
- Defined by RFC 3031, RFC 3032

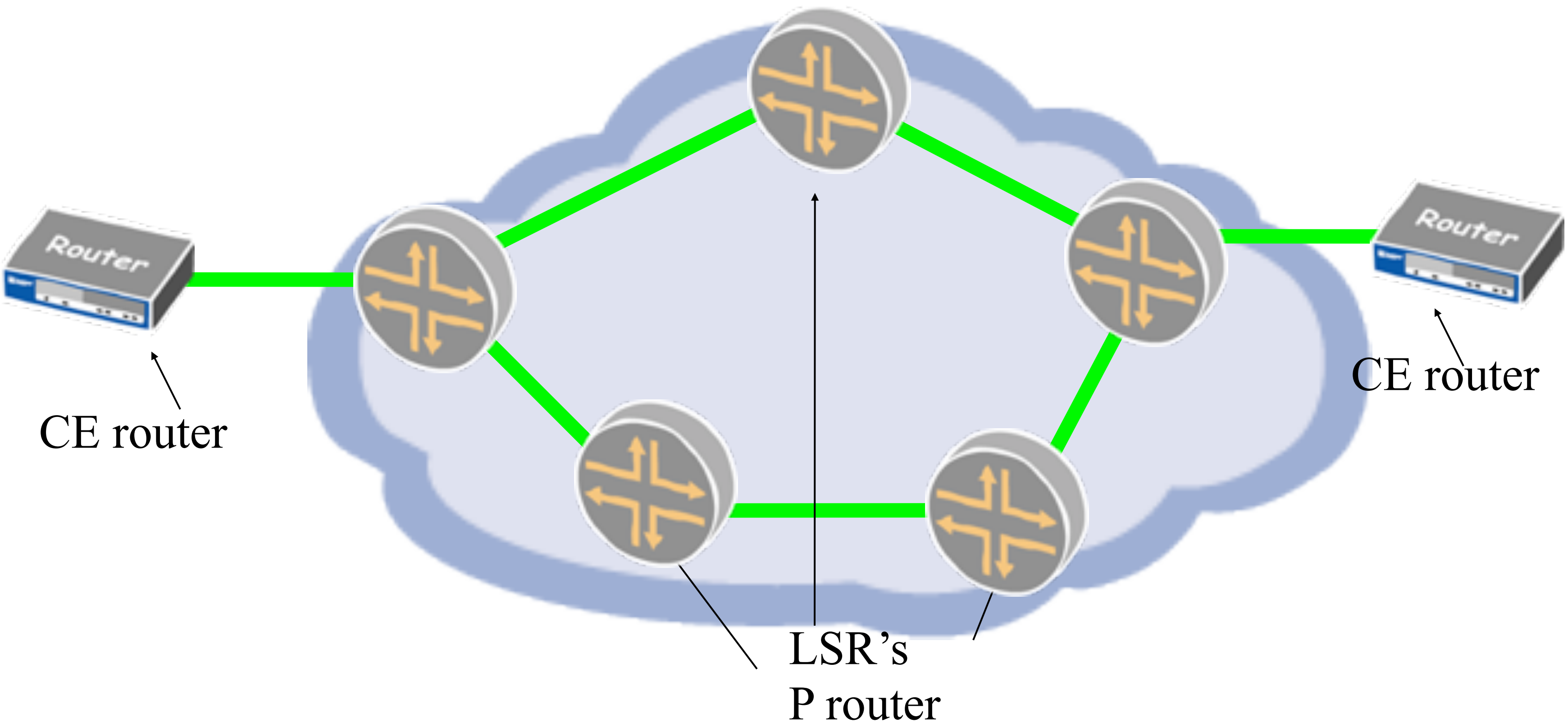
MPLS Terminology



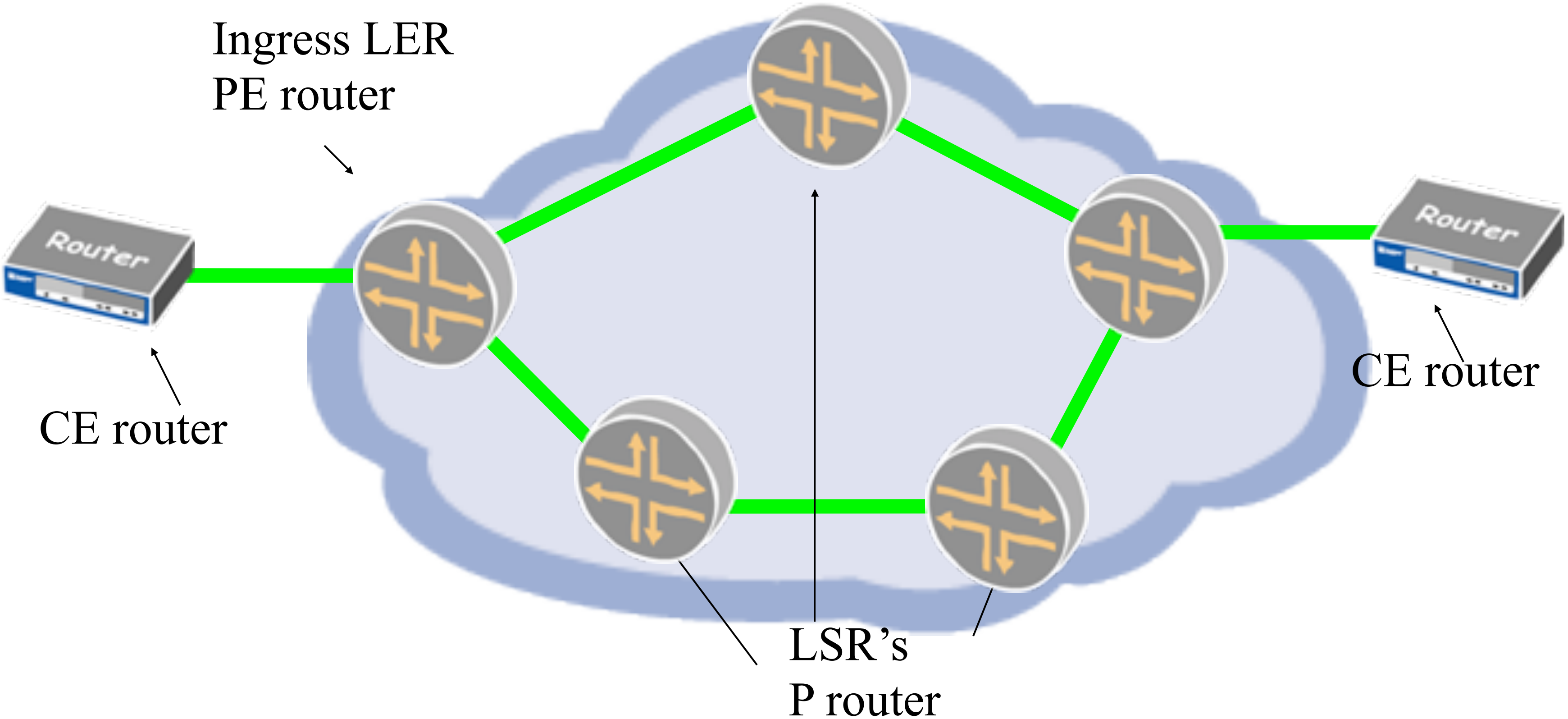
MPLS Terminology



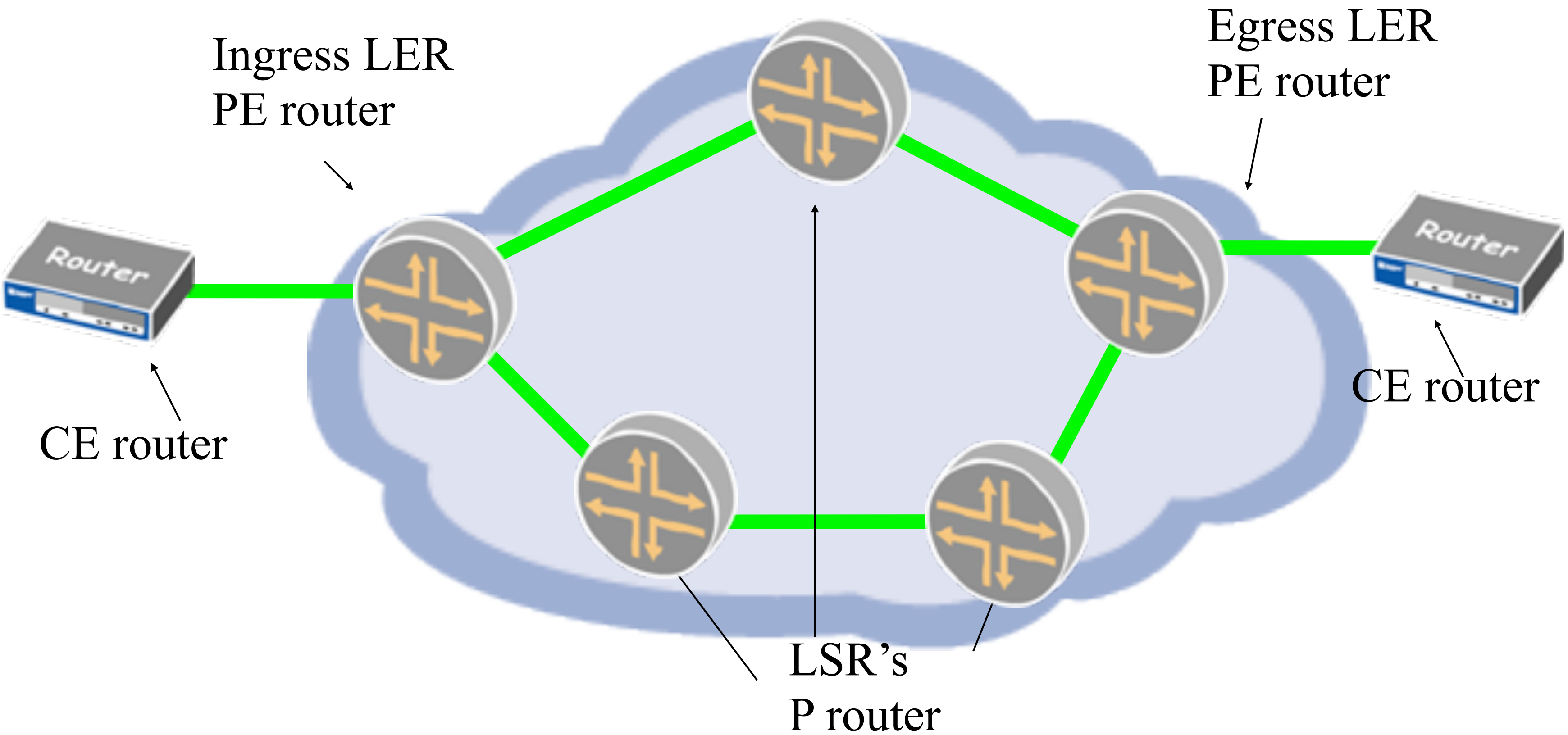
MPLS Terminology



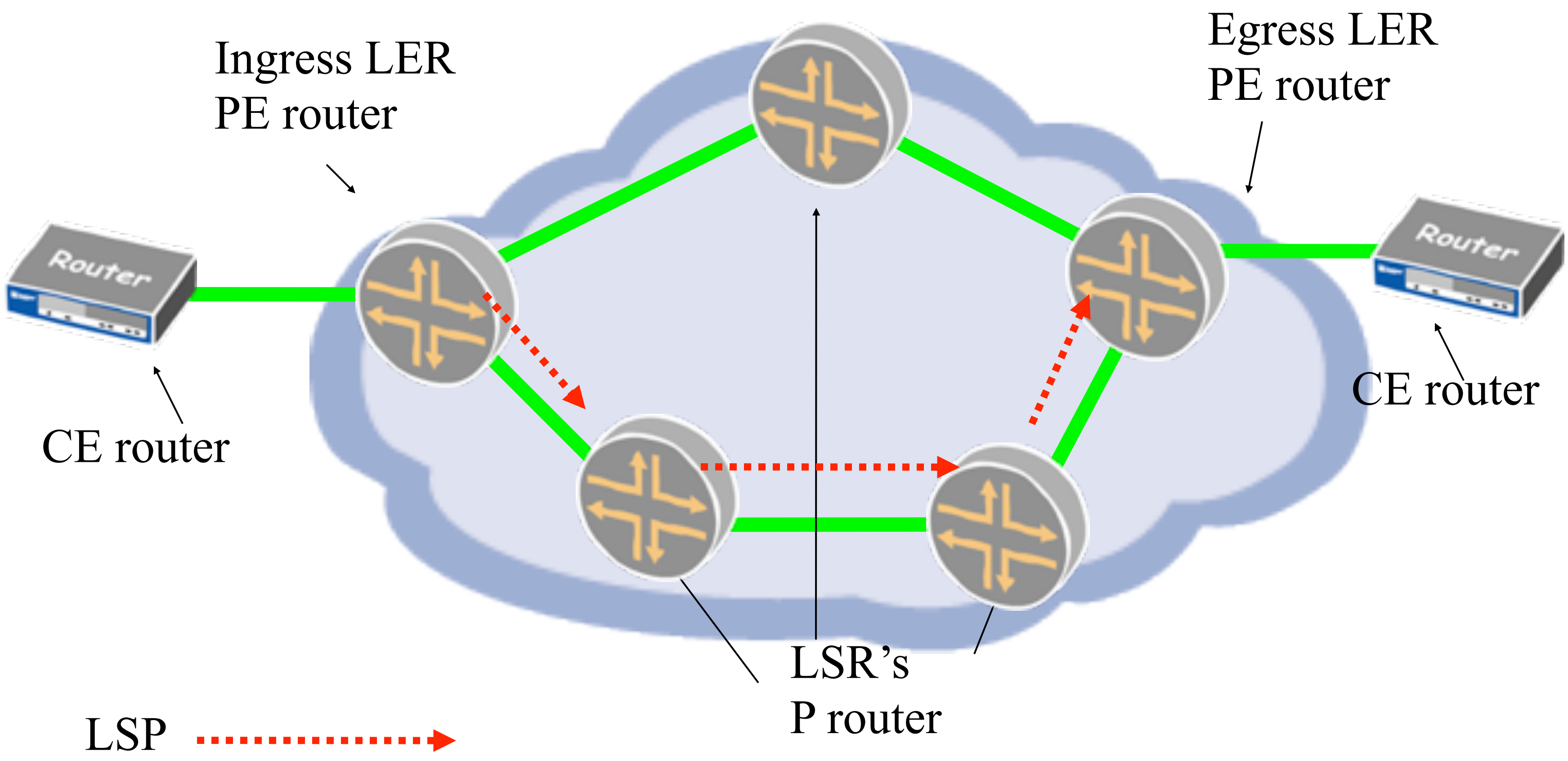
MPLS Terminology



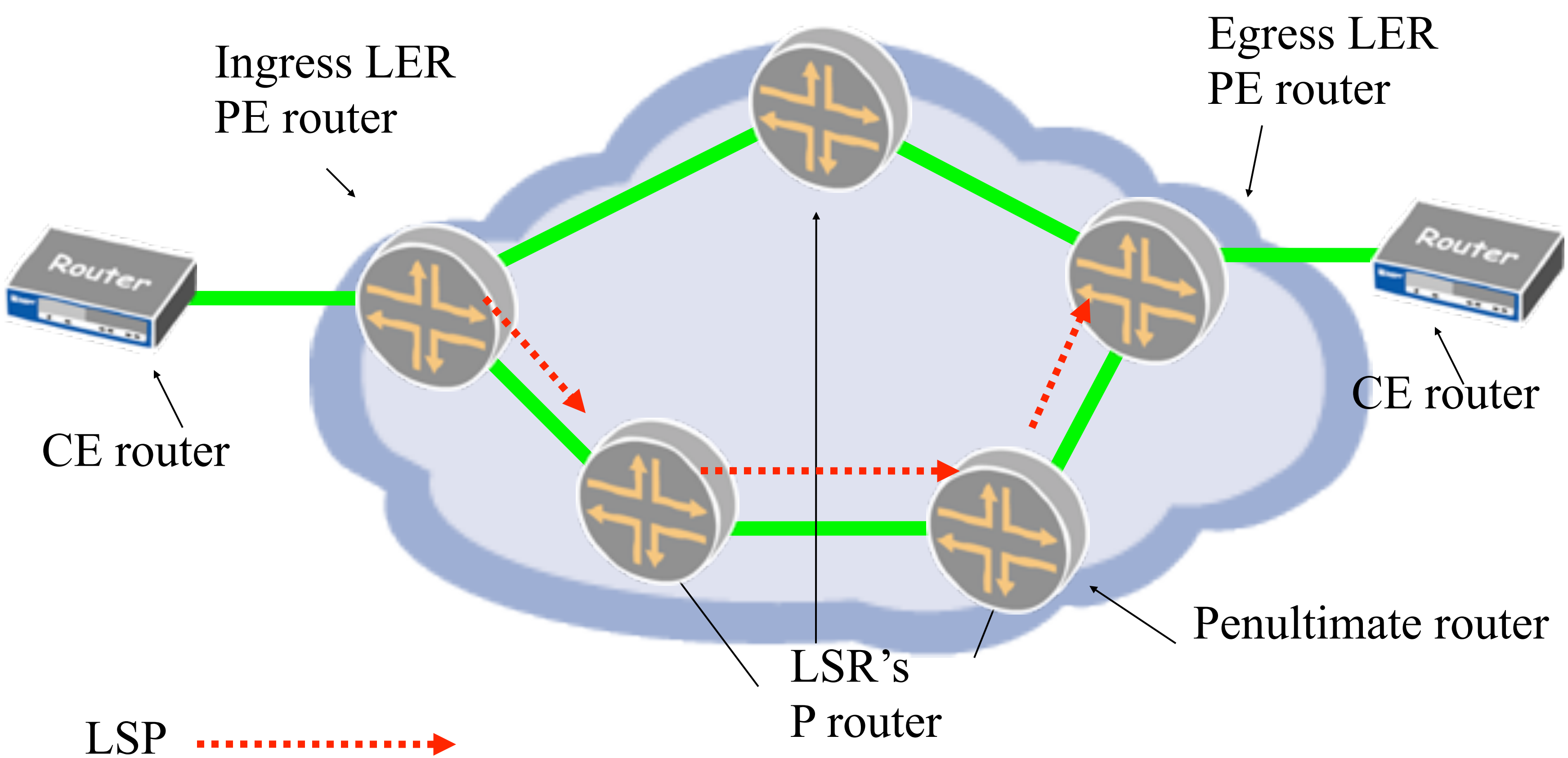
MPLS Terminology



MPLS Terminology

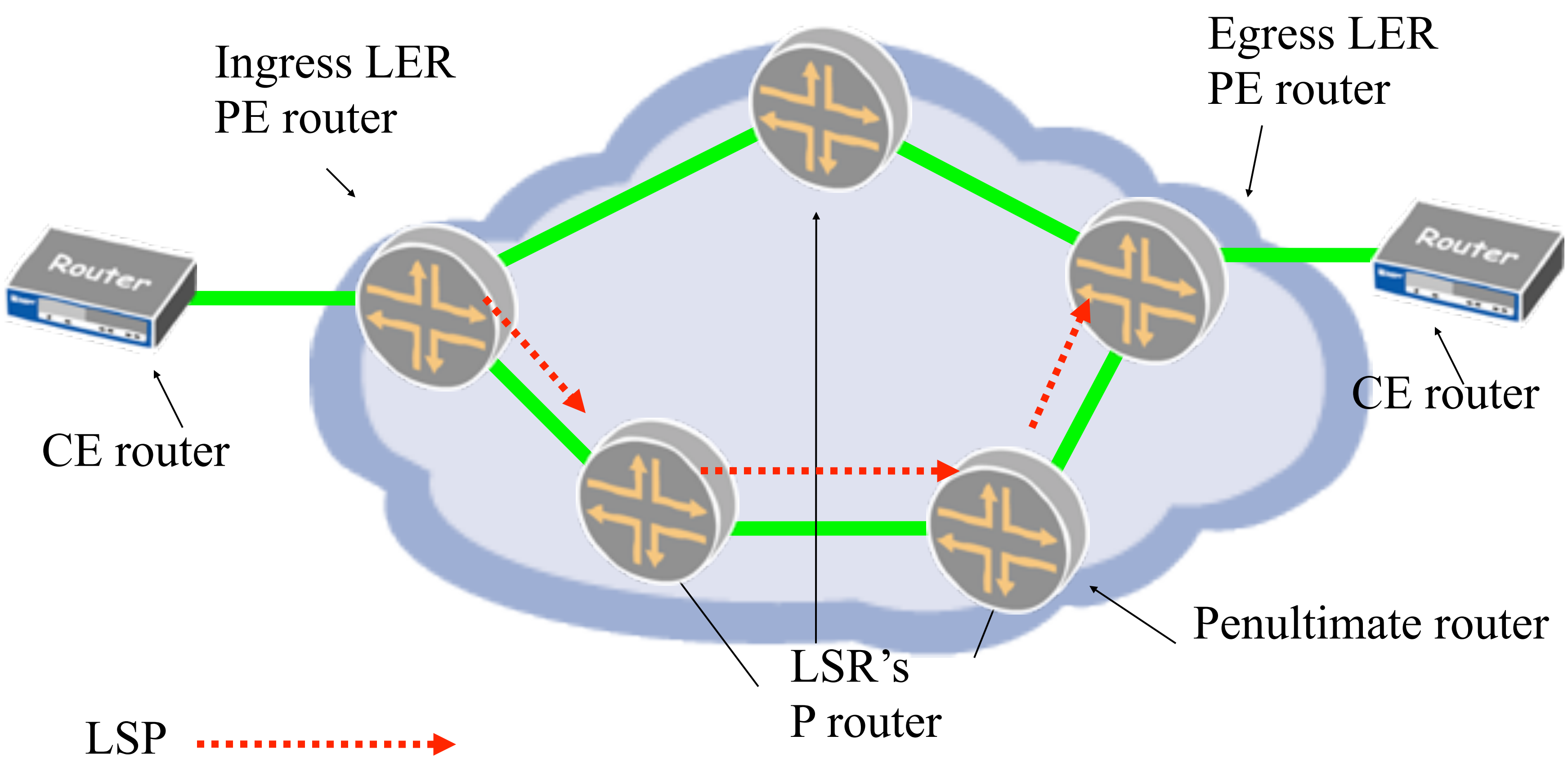


MPLS Terminology



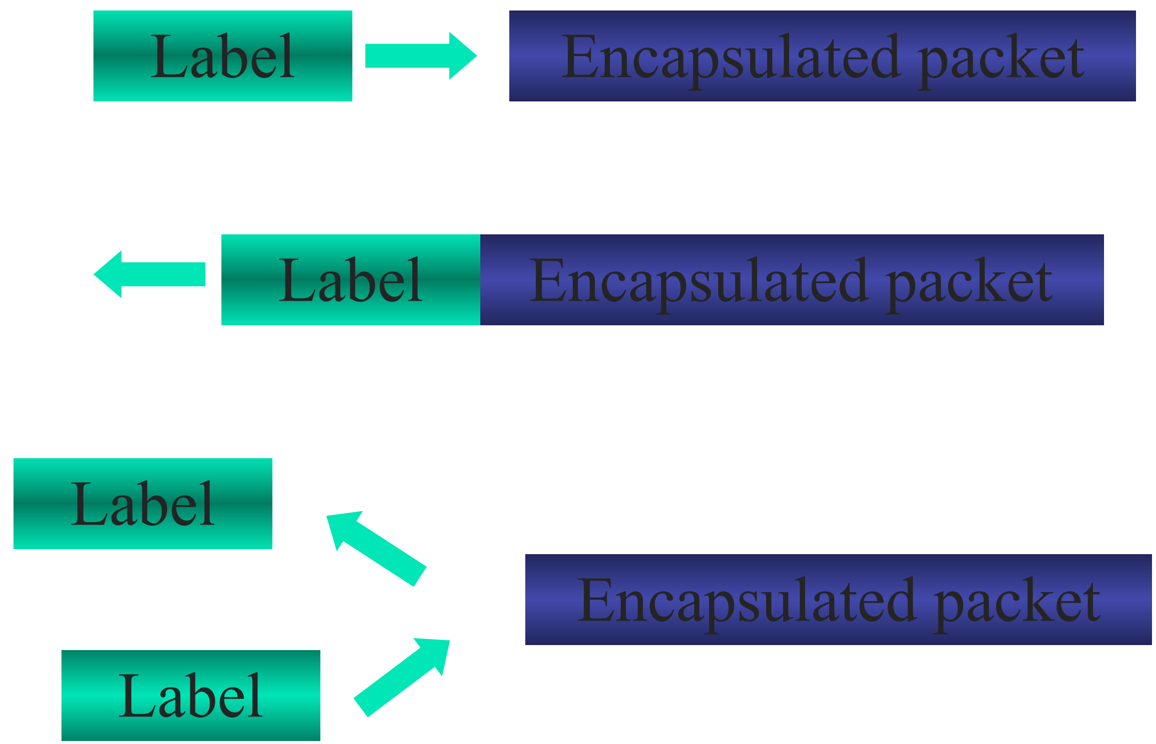
MPLS Terminology

-An LSP is a unidirectional flow of traffic



Push, Pop, Swap

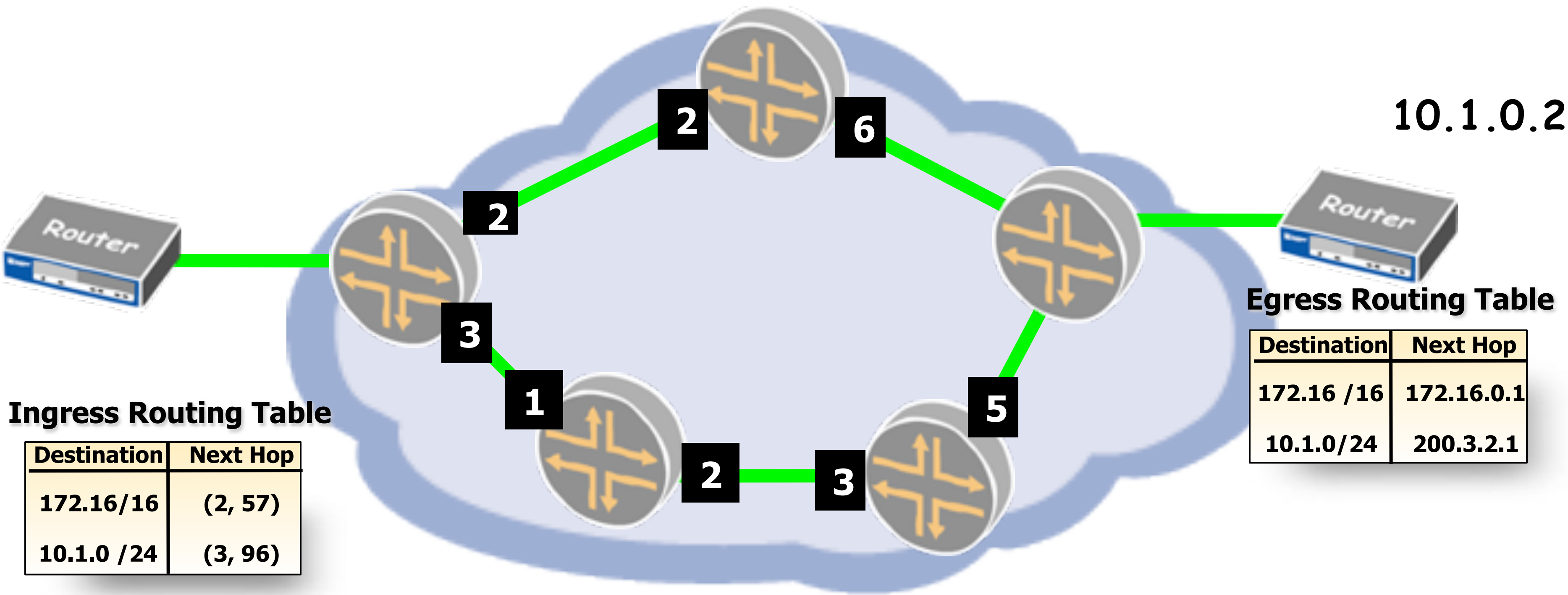
- Push
- Pop
- Swap



MPLS Forwarding Plane

MPLS Table

In	Out
(2, 57)	(6, 0)



Egress Routing Table

Destination	Next Hop
172.16 /16	172.16.0.1
10.1.0/24	200.3.2.1

Ingress Routing Table

Destination	Next Hop
172.16/16	(2, 57)
10.1.0 /24	(3, 96)

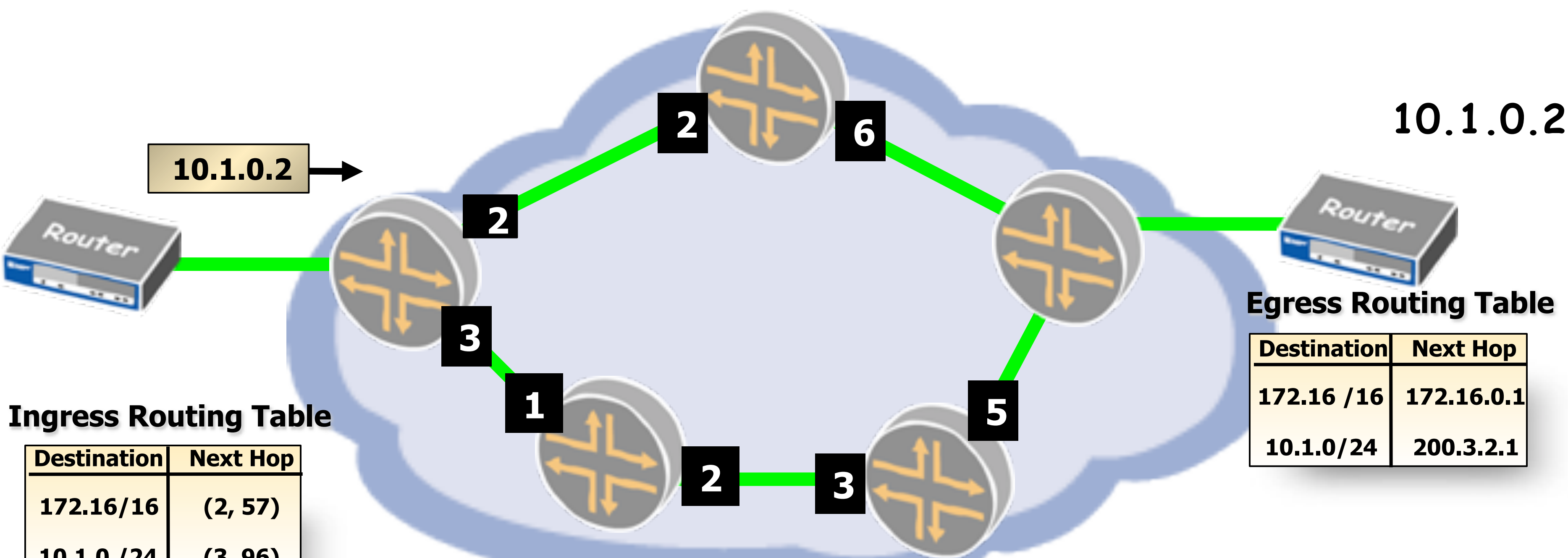
MPLS Table

In	Out
(1, 96)	(2, 32)

MPLS Table

In	Out
(3, 32)	(5, 0)

MPLS Forwarding Plane



MPLS Table

In	Out
(2, 57)	(6, 0)

Ingress Routing Table

Destination	Next Hop
172.16/16	(2, 57)
10.1.0 /24	(3, 96)

Egress Routing Table

Destination	Next Hop
172.16 /16	172.16.0.1
10.1.0/24	200.3.2.1

MPLS Table

In	Out
(1, 96)	(2, 32)

MPLS Table

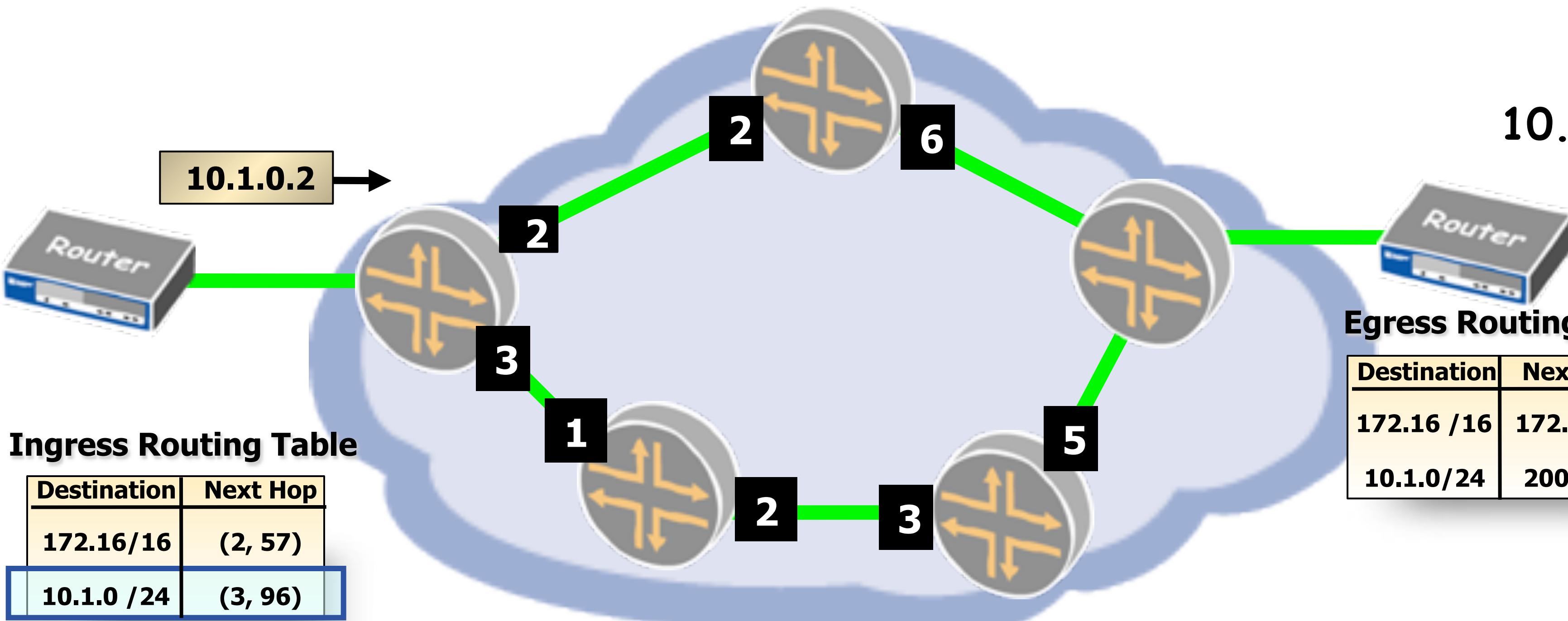
In	Out
(3, 32)	(5, 0)

MPLS Forwarding Plane

MPLS Table

In	Out
(2, 57)	(6, 0)

10.1.0.2



Egress Routing Table

Destination	Next Hop
172.16 /16	172.16.0.1
10.1.0/24	200.3.2.1

Ingress Routing Table

Destination	Next Hop
172.16/16	(2, 57)
10.1.0 /24	(3, 96)

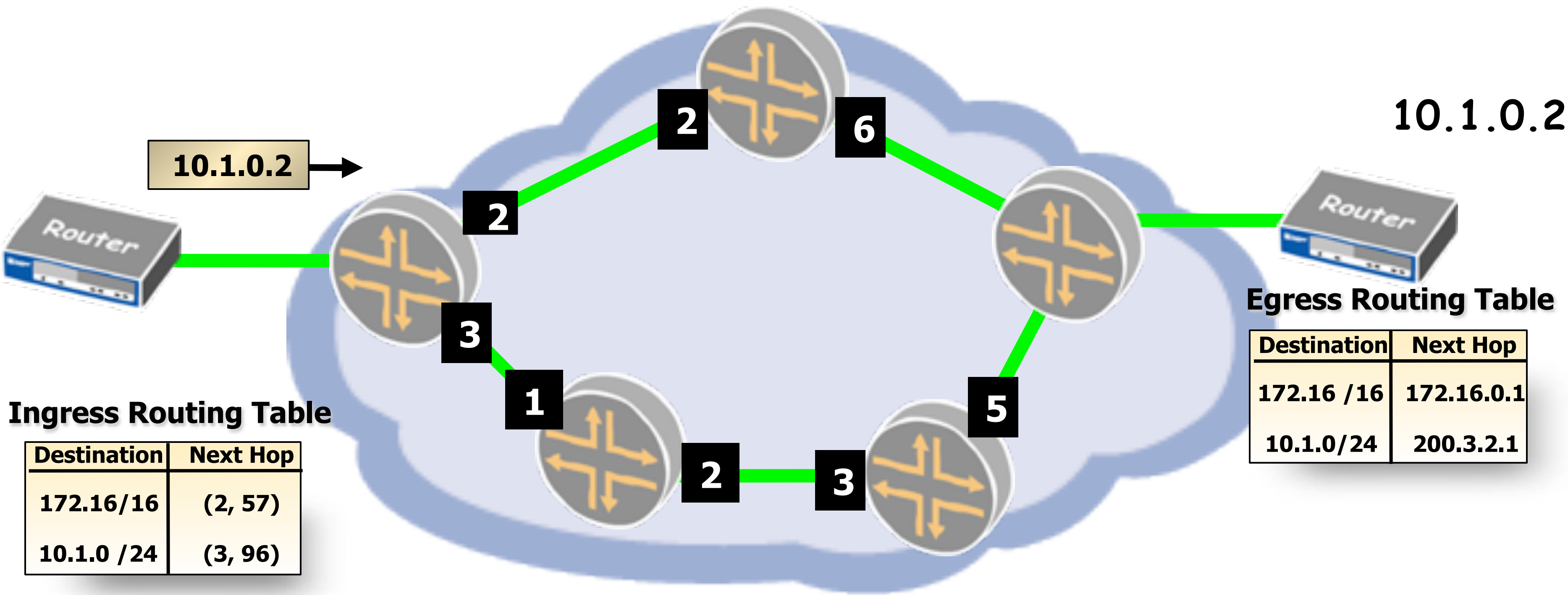
MPLS Table

In	Out
(1, 96)	(2, 32)

MPLS Table

In	Out
(3, 32)	(5, 0)

MPLS Forwarding Plane



MPLS Table

In	Out
(2, 57)	(6, 0)

Ingress Routing Table

Destination	Next Hop
172.16/16	(2, 57)
10.1.0 /24	(3, 96)

Egress Routing Table

Destination	Next Hop
172.16 /16	172.16.0.1
10.1.0/24	200.3.2.1

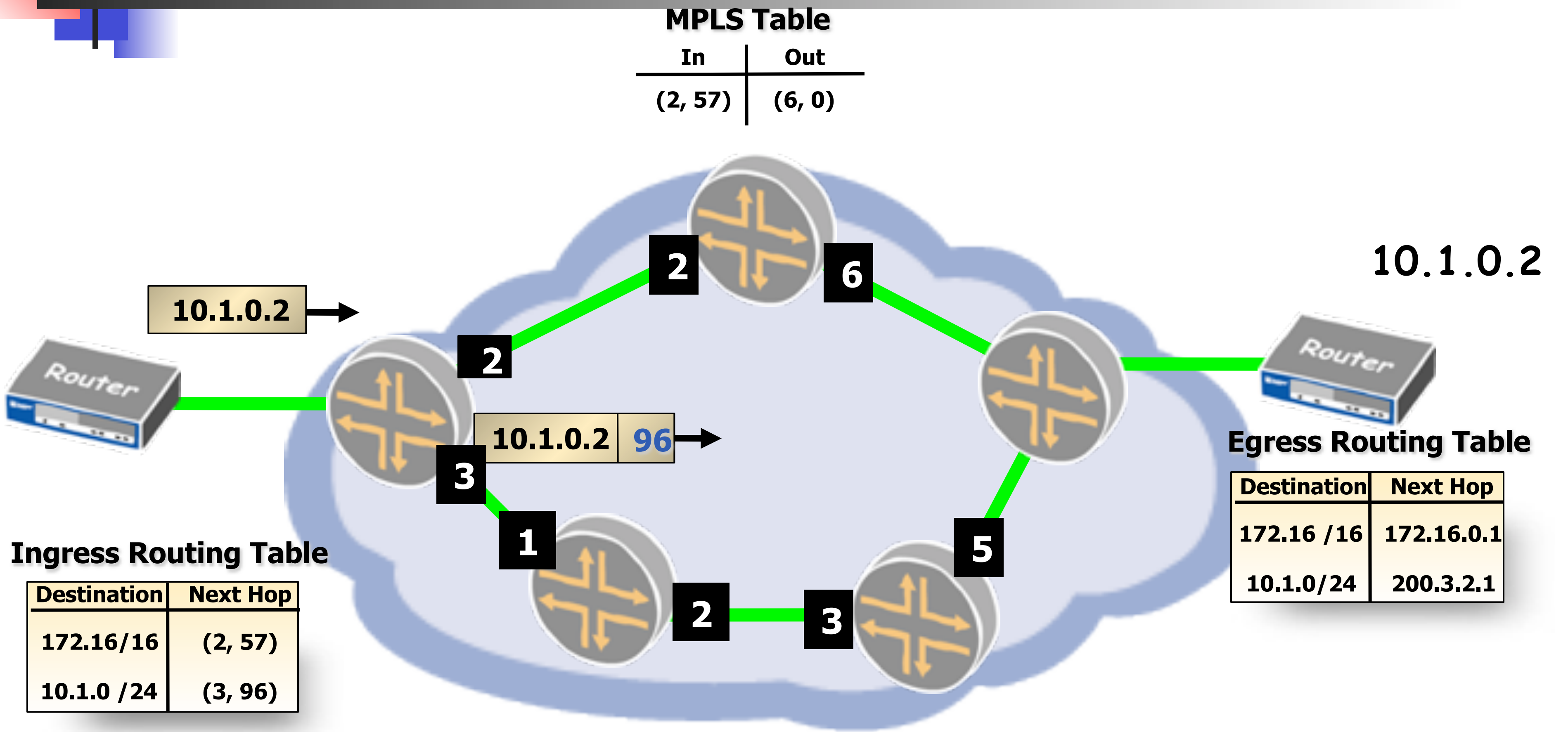
MPLS Table

In	Out
(1, 96)	(2, 32)

MPLS Table

In	Out
(3, 32)	(5, 0)

MPLS Forwarding Plane



Ingress Routing Table

Destination	Next Hop
172.16/16	(2, 57)
10.1.0 /24	(3, 96)

Egress Routing Table

Destination	Next Hop
172.16 /16	172.16.0.1
10.1.0/24	200.3.2.1

MPLS Table

In	Out
(1, 96)	(2, 32)

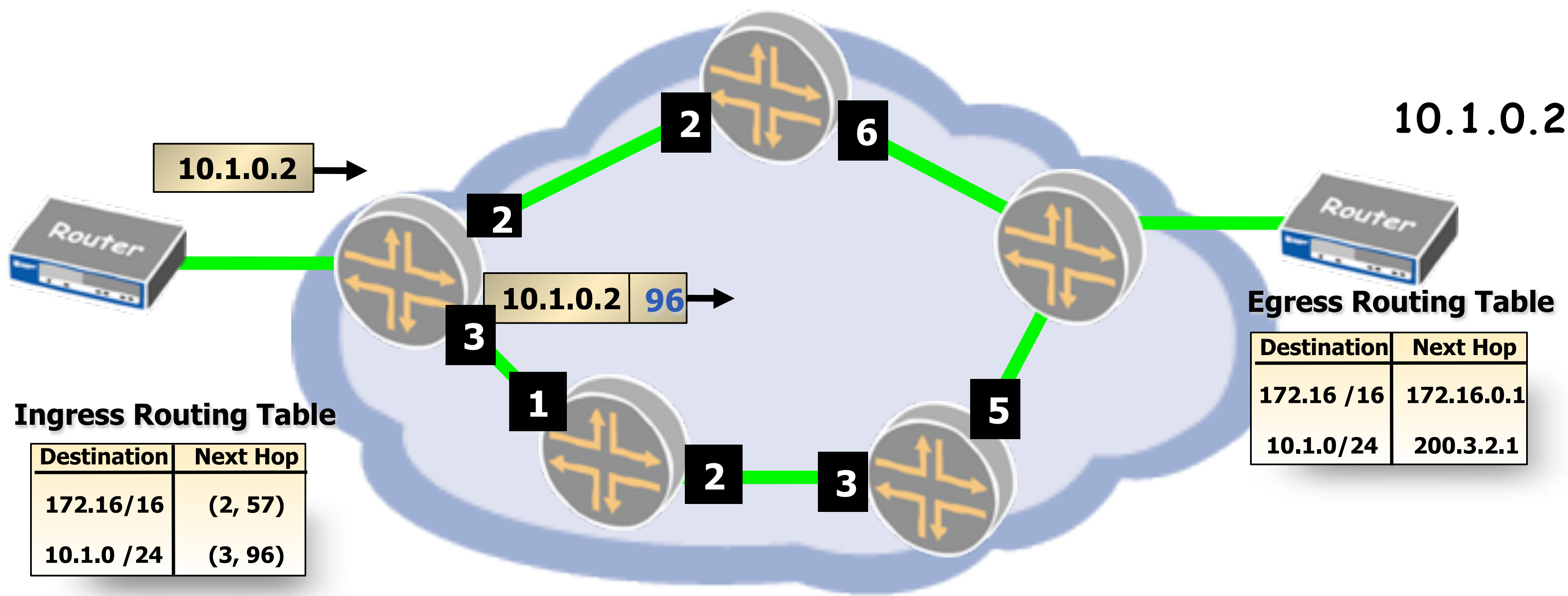
MPLS Table

In	Out
(3, 32)	(5, 0)

MPLS Forwarding Plane

MPLS Table

In	Out
(2, 57)	(6, 0)



Ingress Routing Table

Destination	Next Hop
172.16/16	(2, 57)
10.1.0 /24	(3, 96)

Egress Routing Table

Destination	Next Hop
172.16 /16	172.16.0.1
10.1.0/24	200.3.2.1

MPLS Table

In	Out
(1, 96)	(2, 32)

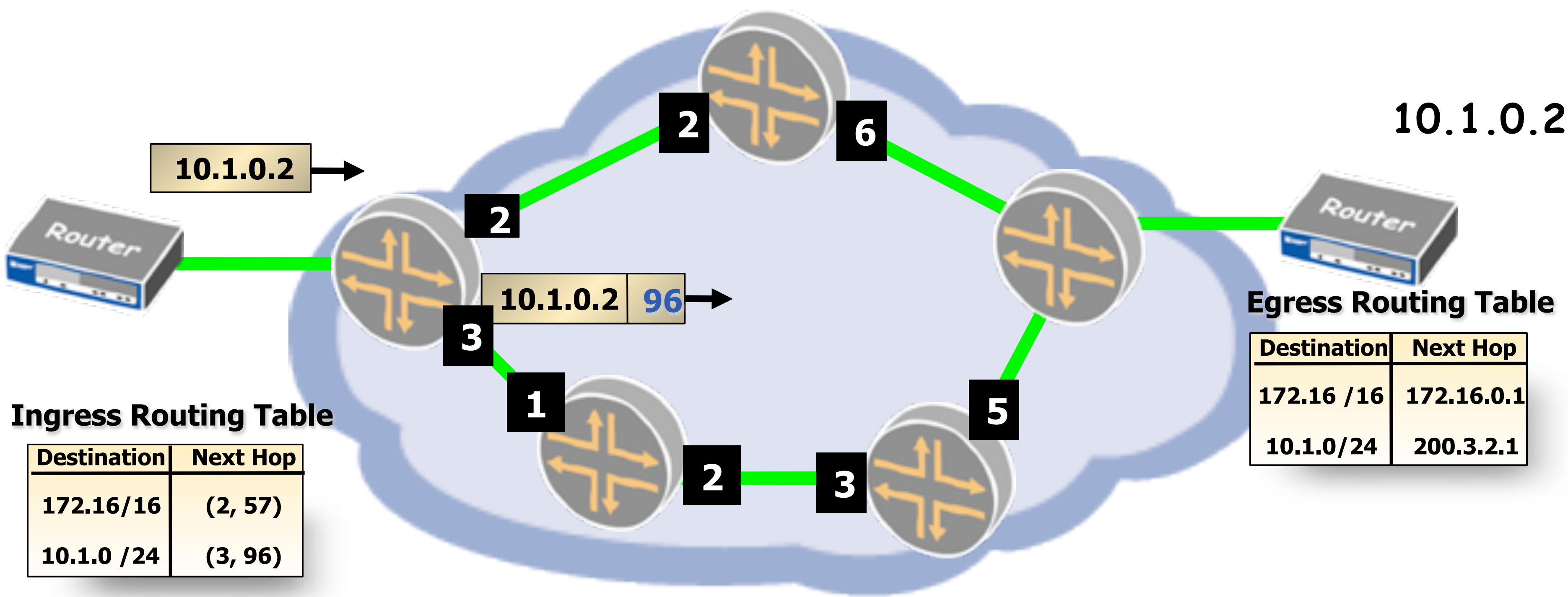
MPLS Table

In	Out
(3, 32)	(5, 0)

MPLS Forwarding Plane

MPLS Table

In	Out
(2, 57)	(6, 0)



Ingress Routing Table

Destination	Next Hop
172.16/16	(2, 57)
10.1.0 /24	(3, 96)

Egress Routing Table

Destination	Next Hop
172.16 /16	172.16.0.1
10.1.0/24	200.3.2.1

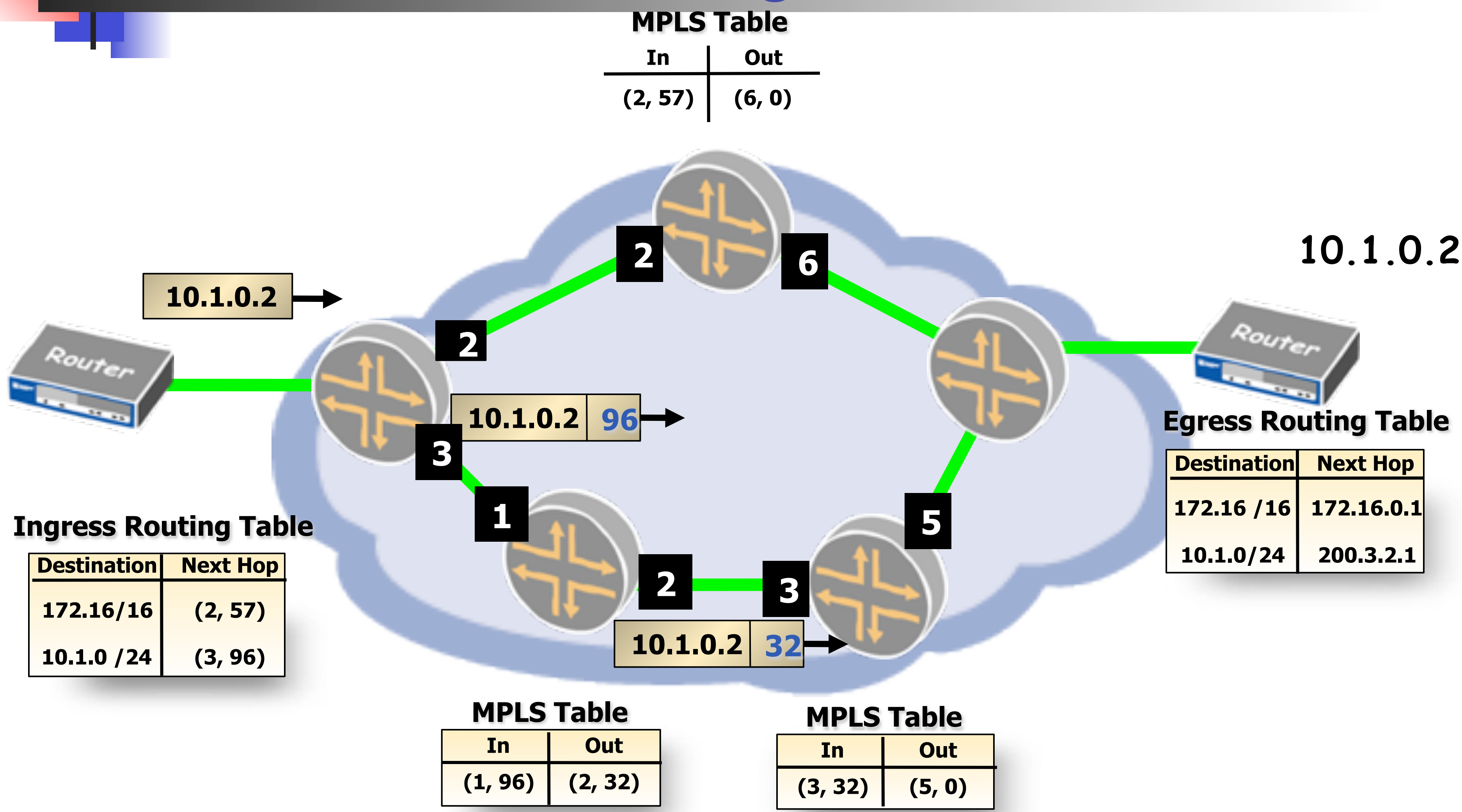
MPLS Table

In	Out
(1, 96)	(2, 32)

MPLS Table

In	Out
(3, 32)	(5, 0)

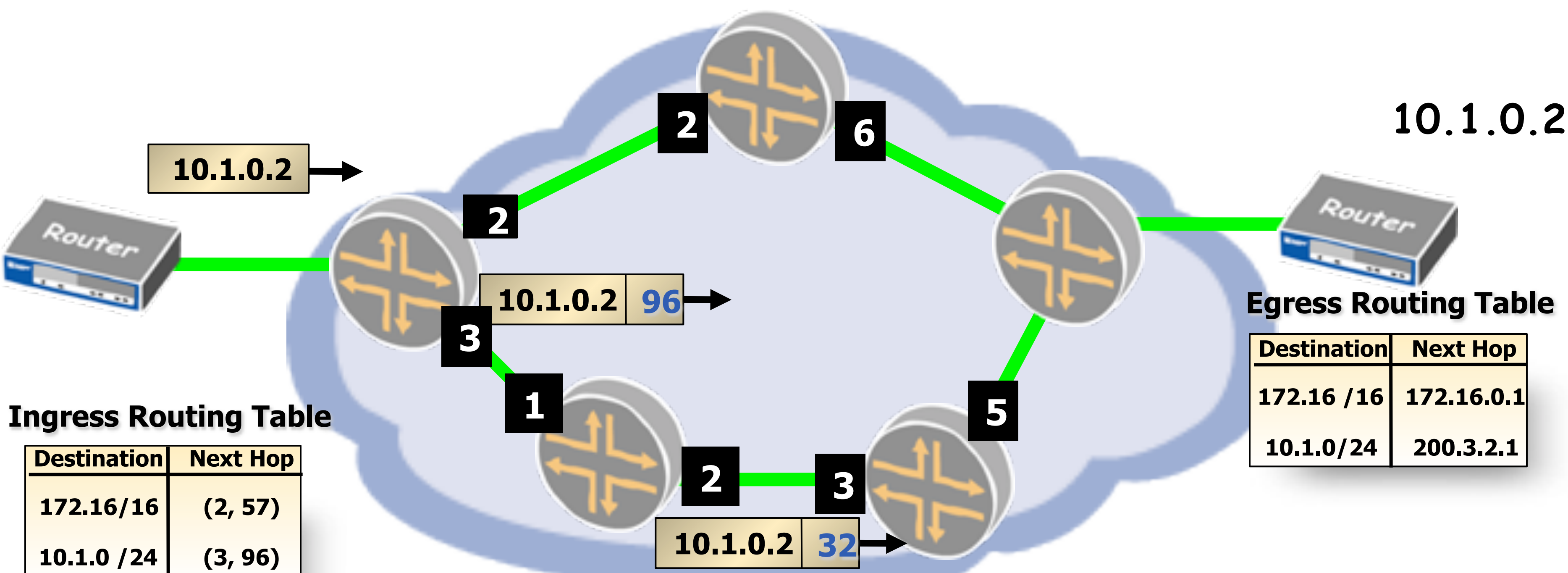
MPLS Forwarding Plane



MPLS Forwarding Plane

MPLS Table

In	Out
(2, 57)	(6, 0)



Egress Routing Table

Destination	Next Hop
172.16 /16	172.16.0.1
10.1.0/24	200.3.2.1

Ingress Routing Table

Destination	Next Hop
172.16/16	(2, 57)
10.1.0 /24	(3, 96)

MPLS Table

In	Out
(1, 96)	(2, 32)

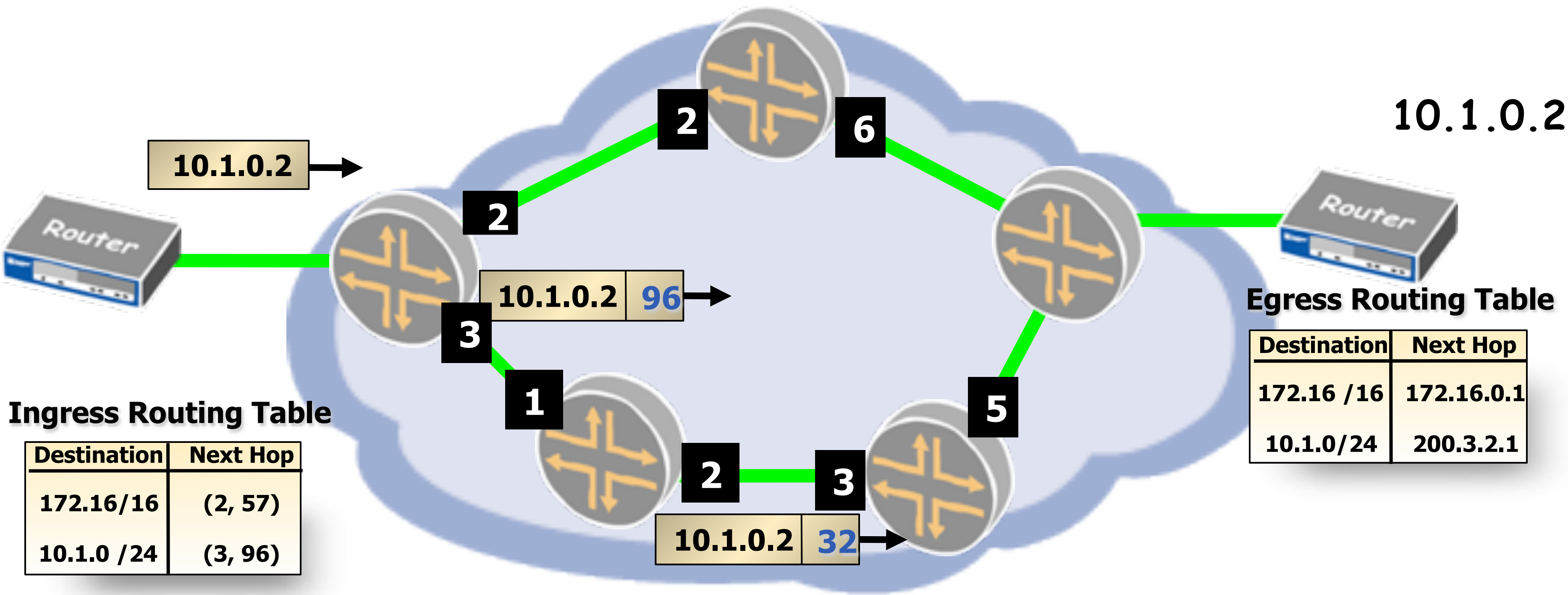
MPLS Table

In	Out
(3, 32)	(5, 0)

MPLS Forwarding Plane

MPLS Table

In	Out
(2, 57)	(6, 0)



Ingress Routing Table

Destination	Next Hop
172.16/16	(2, 57)
10.1.0 /24	(3, 96)

Egress Routing Table

Destination	Next Hop
172.16 /16	172.16.0.1
10.1.0/24	200.3.2.1

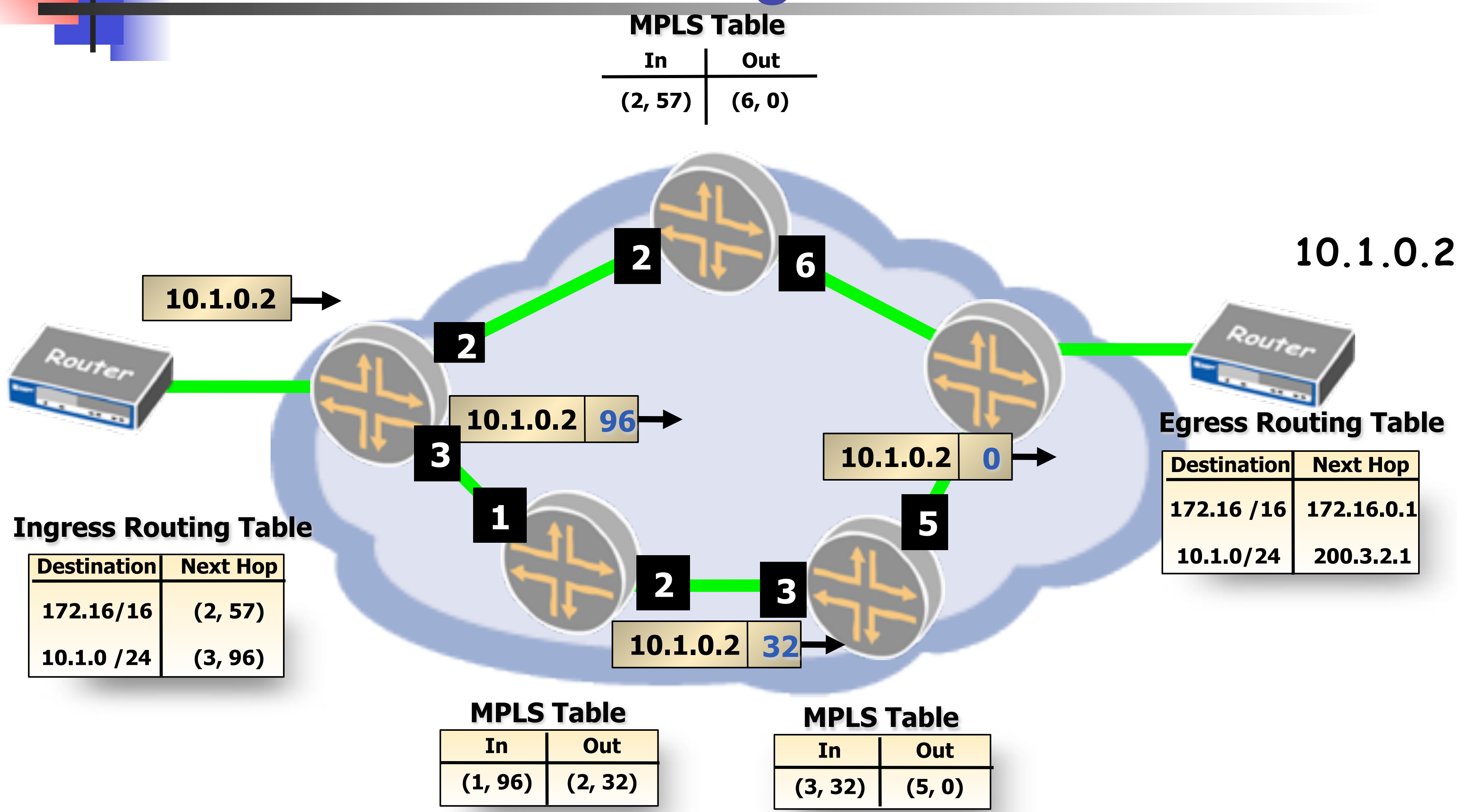
MPLS Table

In	Out
(1, 96)	(2, 32)

MPLS Table

In	Out
(3, 32)	(5, 0)

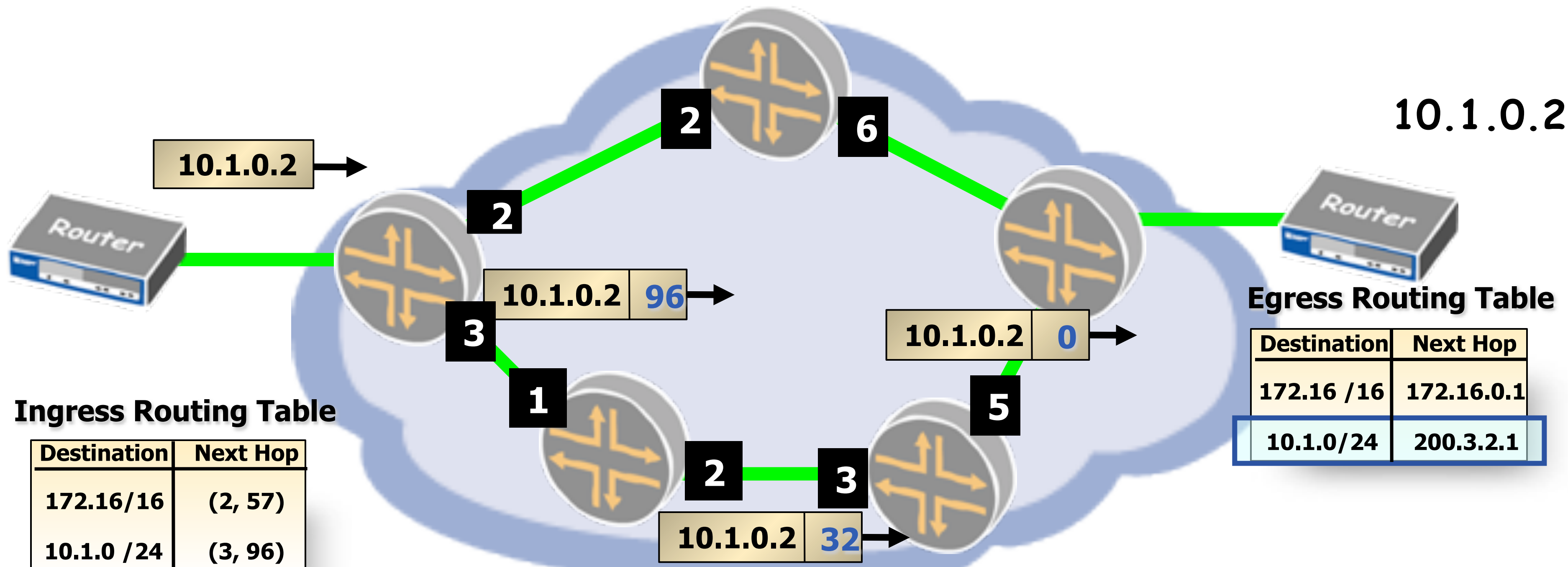
MPLS Forwarding Plane



MPLS Forwarding Plane

MPLS Table

In	Out
(2, 57)	(6, 0)



Ingress Routing Table

Destination	Next Hop
172.16/16	(2, 57)
10.1.0 /24	(3, 96)

Egress Routing Table

Destination	Next Hop
172.16 /16	172.16.0.1
10.1.0/24	200.3.2.1

MPLS Table

In	Out
(1, 96)	(2, 32)

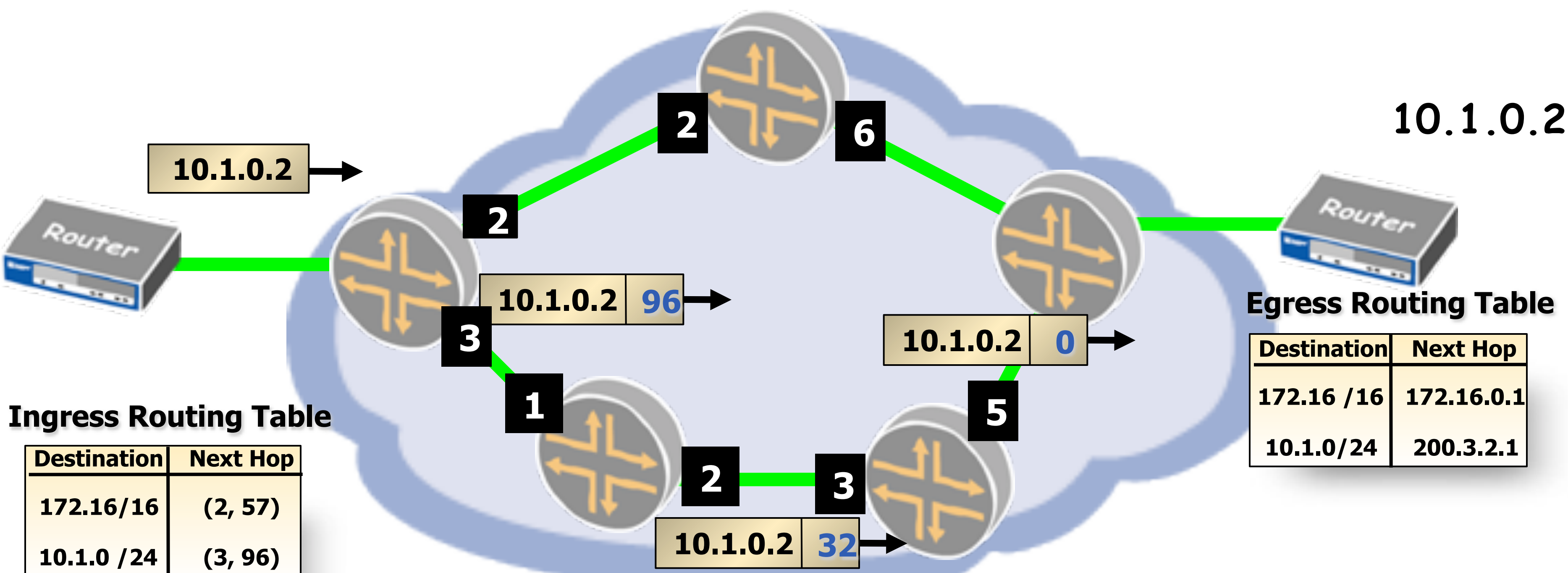
MPLS Table

In	Out
(3, 32)	(5, 0)

MPLS Forwarding Plane

MPLS Table

In	Out
(2, 57)	(6, 0)



Ingress Routing Table

Destination	Next Hop
172.16/16	(2, 57)
10.1.0 /24	(3, 96)

Egress Routing Table

Destination	Next Hop
172.16 /16	172.16.0.1
10.1.0/24	200.3.2.1

MPLS Table

In	Out
(1, 96)	(2, 32)

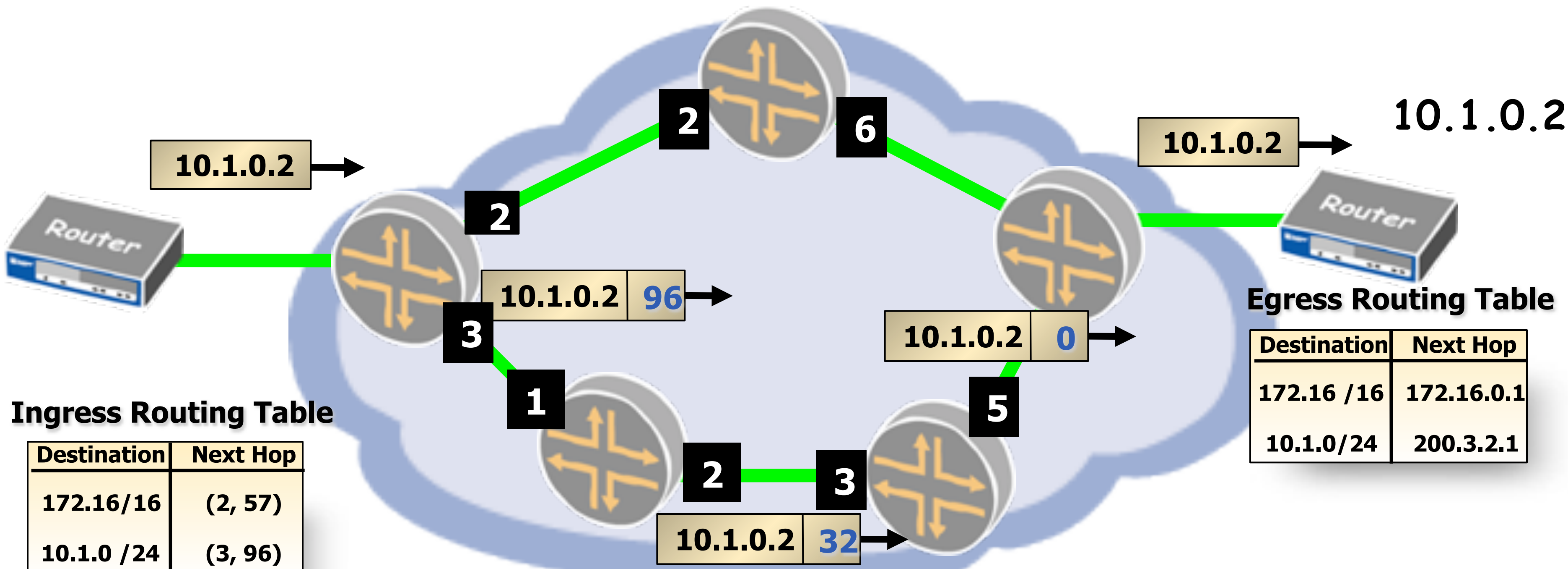
MPLS Table

In	Out
(3, 32)	(5, 0)

MPLS Forwarding Plane

MPLS Table

In	Out
(2, 57)	(6, 0)



Ingress Routing Table

Destination	Next Hop
172.16/16	(2, 57)
10.1.0 /24	(3, 96)

Egress Routing Table

Destination	Next Hop
172.16 /16	172.16.0.1
10.1.0/24	200.3.2.1

MPLS Table

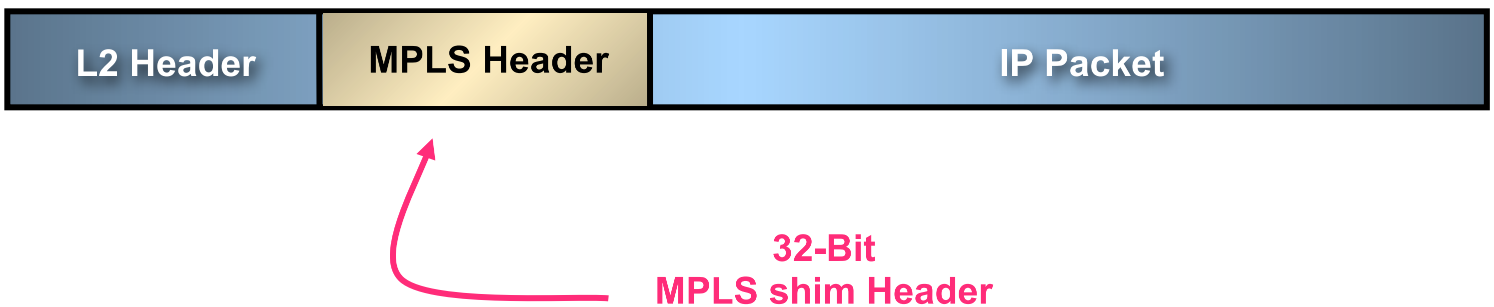
In	Out
(1, 96)	(2, 32)

MPLS Table

In	Out
(3, 32)	(5, 0)

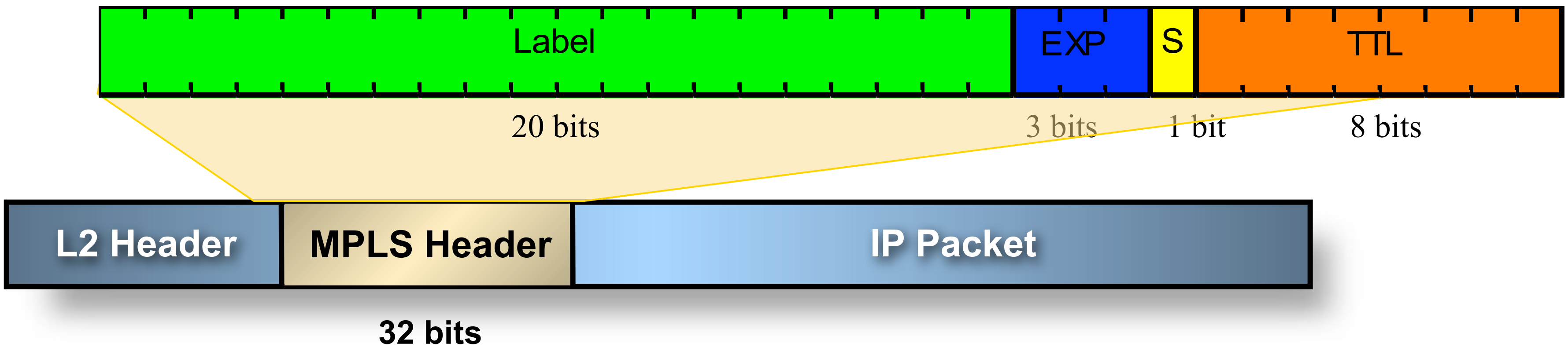
Labeled Packets

- MPLS header is prepended to packet with a push operation at ingress node
  - Label is added immediately after Layer 2 encapsulation



- Packet is restored at the end of the LSP with a pop operation
  - Normally the label stack is popped at penultimate node

The Label



- Label
  - Used to identify virtual circuit
- EXP
  - Experimental. Currently this is used to identify class of servce (CoS)
- S (Stack Bit)
  - Used to indicate if there is another label inside this packet or is it the original encapsulated data
- TTL
  - Time to live, functionally equivalent to IP TTL.

Example - Ethernet

Example - Ethernet

0 0 1 0 1 1 1 1 0 1 0 0 0 1 0 1 1

Example - Ethernet

0 0 1 0 1 1 1 1 0 1 0 0 0 1 0 1 1

Dest. MAC

Example - Ethernet

0 0 1 0 1 1 1 1 0 1 0 0 0 1 0 1 1

Dest. MAC Src. MAC

Example - Ethernet

0 0 1 0 1 1 1 1 0 1 0 0 0 1 0 1 1

Dest. MAC	Src. MAC	Type = 8347

Example - Ethernet

0 0 1 0 1 1 1 1 0 1 0 0 0 1 0 1 1

Label = 47 | EXP = BE | S = 1 | TTL = 240

Dest. MAC   Src. MAC   Type = 8347

Example - Ethernet

0 0 1 0 1 1 1 1 0 1 0 0 0 1 0 1 1

Label = 23 | EXP = BE | S = 0 | TTL = 254

Label = 47 | EXP = BE | S = 1 | TTL = 240

Dest. MAC Src. MAC Type = 8347

Example - Ethernet

0 0 1 0 1 1 1 1 0 1 0 0 0 1 0 1 1

IP Header | Protocol = TCP
Label = 23 | EXP = BE | S = 0 | TTL = 254
Label = 47 | EXP = BE | S = 1 | TTL = 240
Dest. MAC Src. MAC Type = 8347

Example - Ethernet

0 0 1 0 1 1 1 1 0 1 0 0 0 1 0 1 1

TCP | port = 80 (www)

IP Header | Protocol = TCP

Label = 23 | EXP = BE | S = 0 | TTL = 254

Label = 47 | EXP = BE | S = 1 | TTL = 240

Dest. MAC Src. MAC Type = 8347

Example - Ethernet

0 0 1 0 1 1 1 1 0 1 0 0 0 1 0 1 1

My Web Page

TCP | port = 80 (www)

IP Header | Protocol = TCP

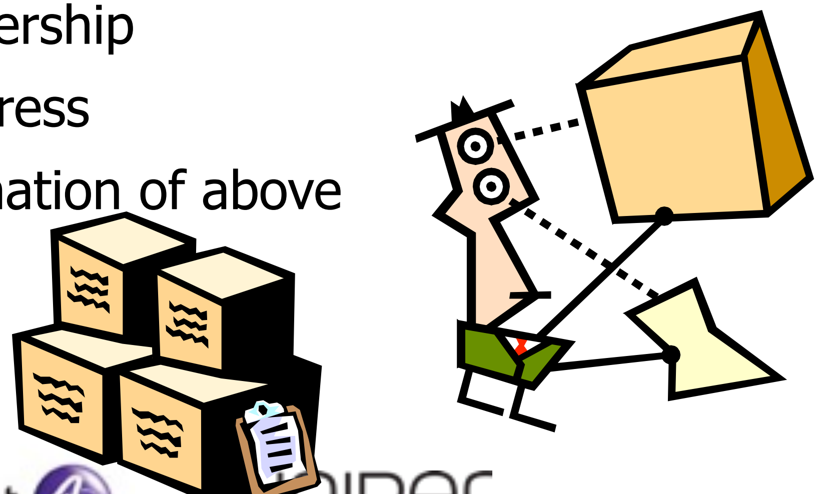
Label = 23 | EXP = BE | S = 0 | TTL = 254

Label = 47 | EXP = BE | S = 1 | TTL = 240

Dest. MAC Src. MAC Type = 8347

FEC – Forwarding Equivalency class

- All traffic with the same FEC will follow the same path and experience same level of service
- E.g. of FEC
  - Destination IP address
  - BGP next hop
  - VPN membership
  - Source address
  - Any combination of above



FEC – Forwarding Equivalency class

- All traffic with the same FEC will follow the same path and experience same level of service
- E.g. of FEC
  - Destination IP address
  - BGP next hop
  - VPN membership
  - Source address
  - Any combination of above

FEC – Forwarding Equivalency class

- All traffic with the same FEC will follow the same path and experience same level of service
- E.g. of FEC
  - Destination IP address
  - BGP next hop
  - VPN membership
  - Source address
  - Any combination of above

What label is pushed onto what packet?

Packet

Label

Signaling

- Protocols that are used to setup maintain and tear down LSP's.
- Can behave differently depending on function
- Let's describe a language / concepts to understand these differences in operation

Signaling

- Protocols that are used to setup maintain and tear down LSP's.
- Can behave differently depending on function
- Let's describe a language / concepts to understand these differences in operation

Signalling Protocols

- LDP
  - Label Distribution Protocol
- RSVP-TE
  - Resource Reservation Protocol with Traffic Engineering Extensions
- MBGP
  - Multi-protocol BGP

Signalling Protocols

- LDP
  - Label Distribution Protocol
- RSVP-TE
  - Resource Reservation Protocol with Traffic Engineering Extensions
- MBGP
  - Multi-protocol BGP

Which to choose…

- Traffic Engineering, Traffic Protection
  - RSVP
  - Link State protocol
- VPN's
  - LDP or RSVP (all LSR's)
  - MBGP (PE's only)
- Why use LDP at all?
  - Configuration scaling
  - LDP configuration is "per box"
  - RSVP configuration is "per LSP"

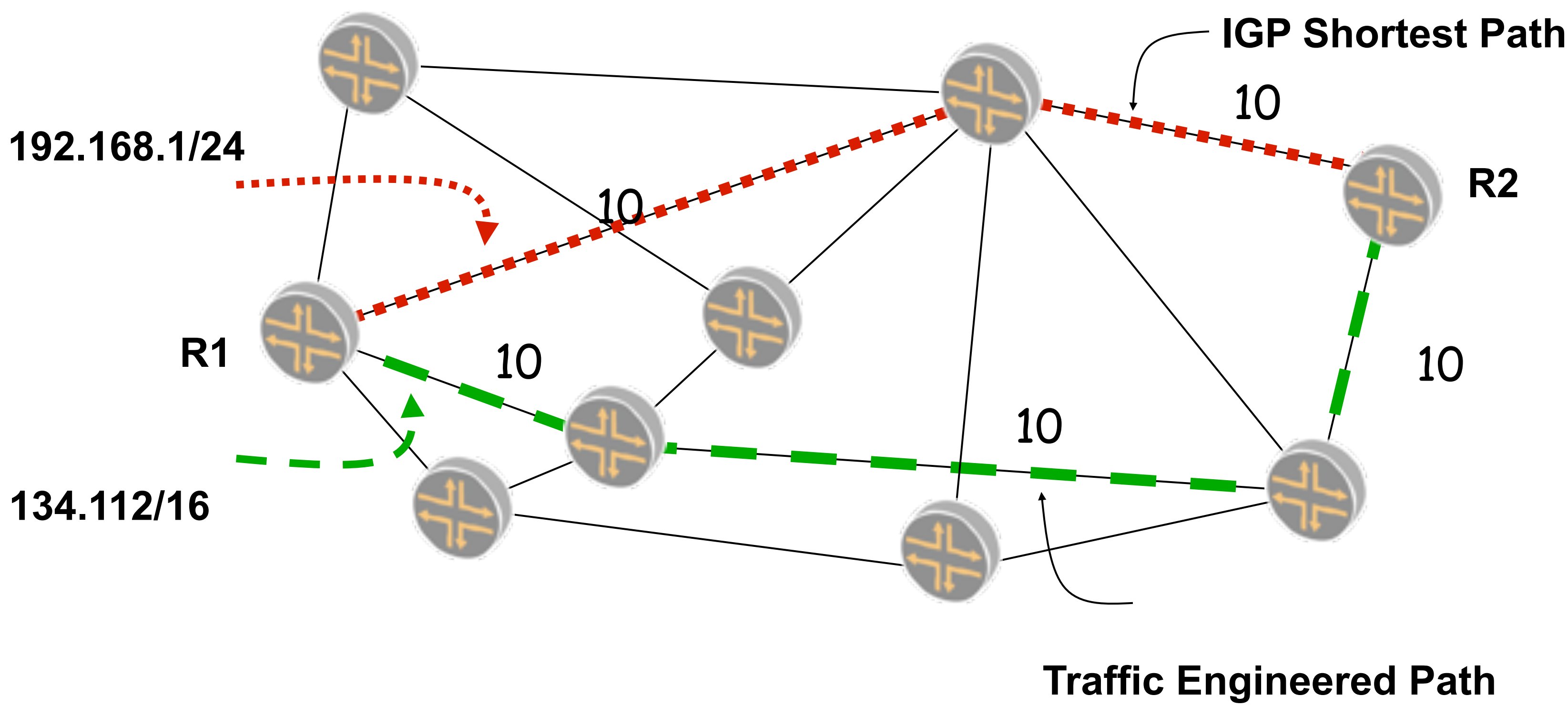
Which to choose…

- Traffic Engineering, Traffic Protection
  - RSVP
  - Link State protocol
- VPN's
  - LDP or RSVP (all LSR's)
  - MBGP (PE's only)
- Why use LDP at all?
  - Configuration scaling
  - LDP configuration is "per box"
  - RSVP configuration is "per LSP"

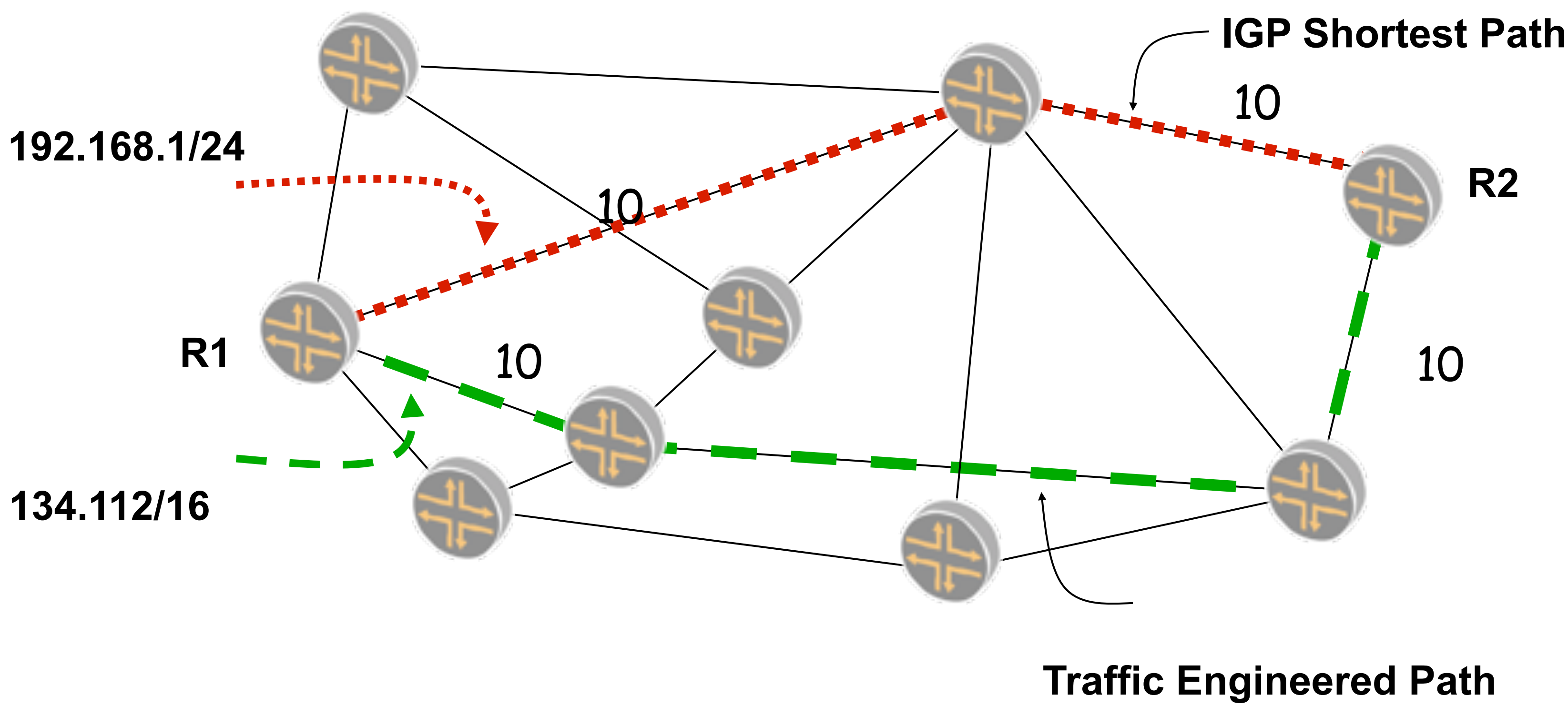
Traffic Engineering Defined

- Sub Optimal routing
- Network Engineering is putting bandwidth where the traffic is. Traffic Engineering is putting the traffic where the bandwidth is!
- To meet one of two requirements
  - To better utilize network capacity and resources.
  - To put traffic on a path that can support it's requirements
- Incorporate Traffic Protection to achieve SONET like failure recovery.

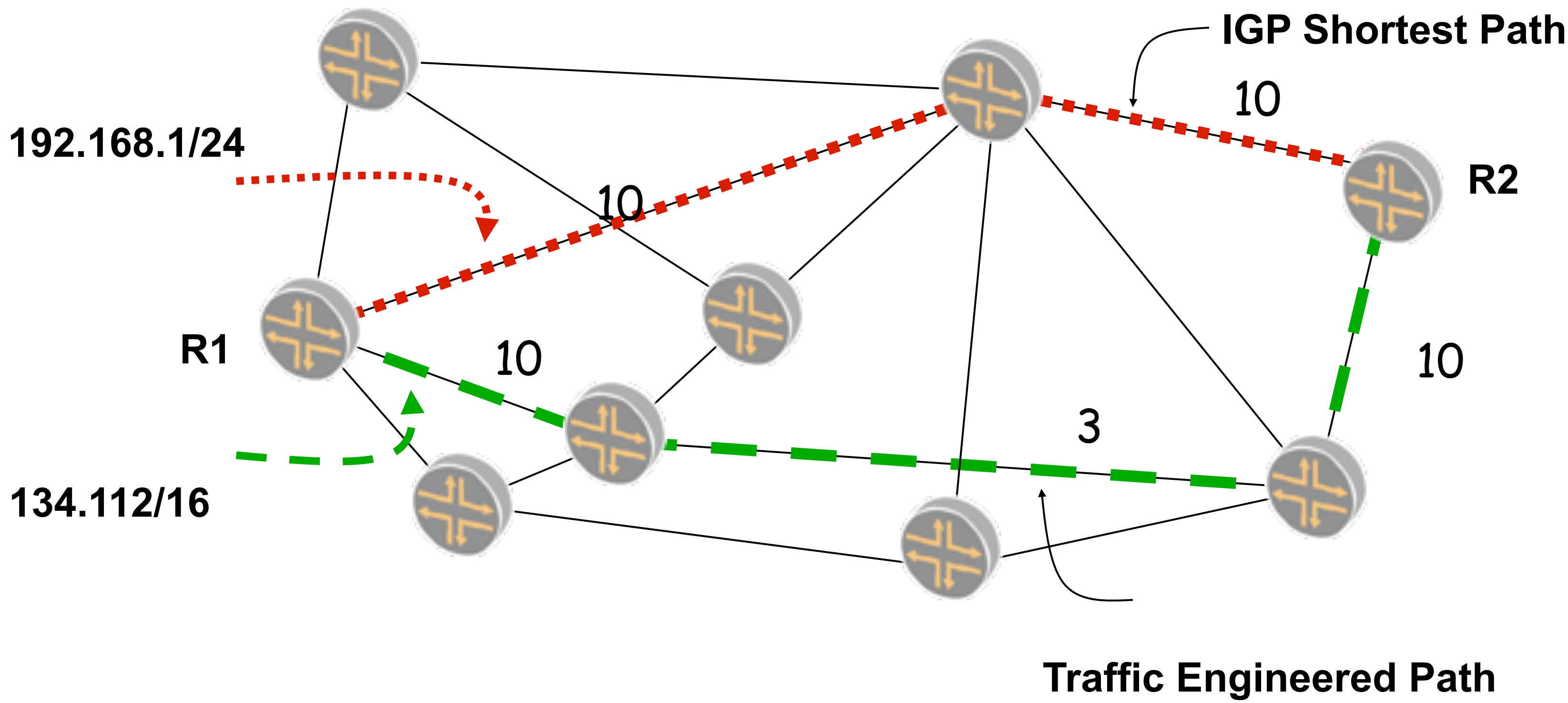
MPLS-Based Traffic Engineering



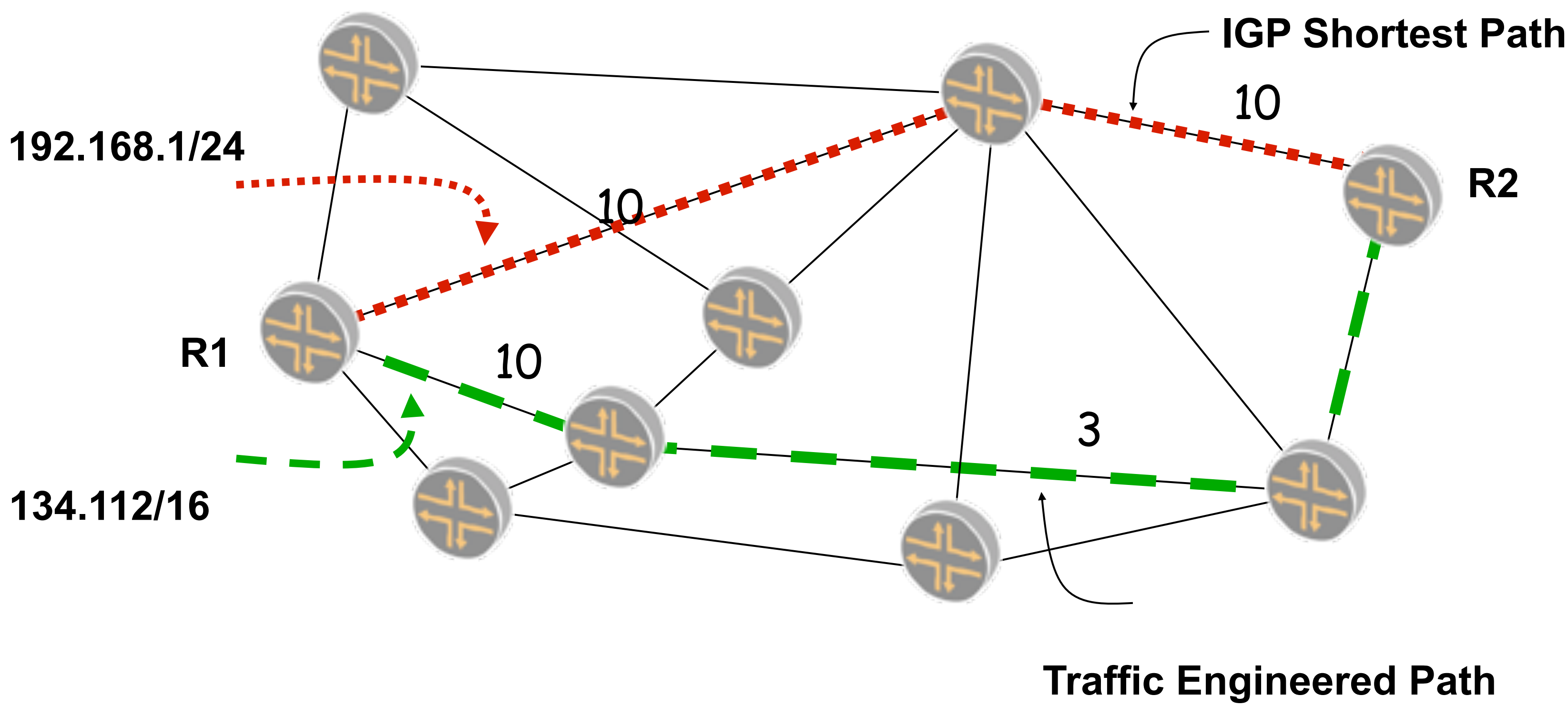
MPLS-Based Traffic Engineering



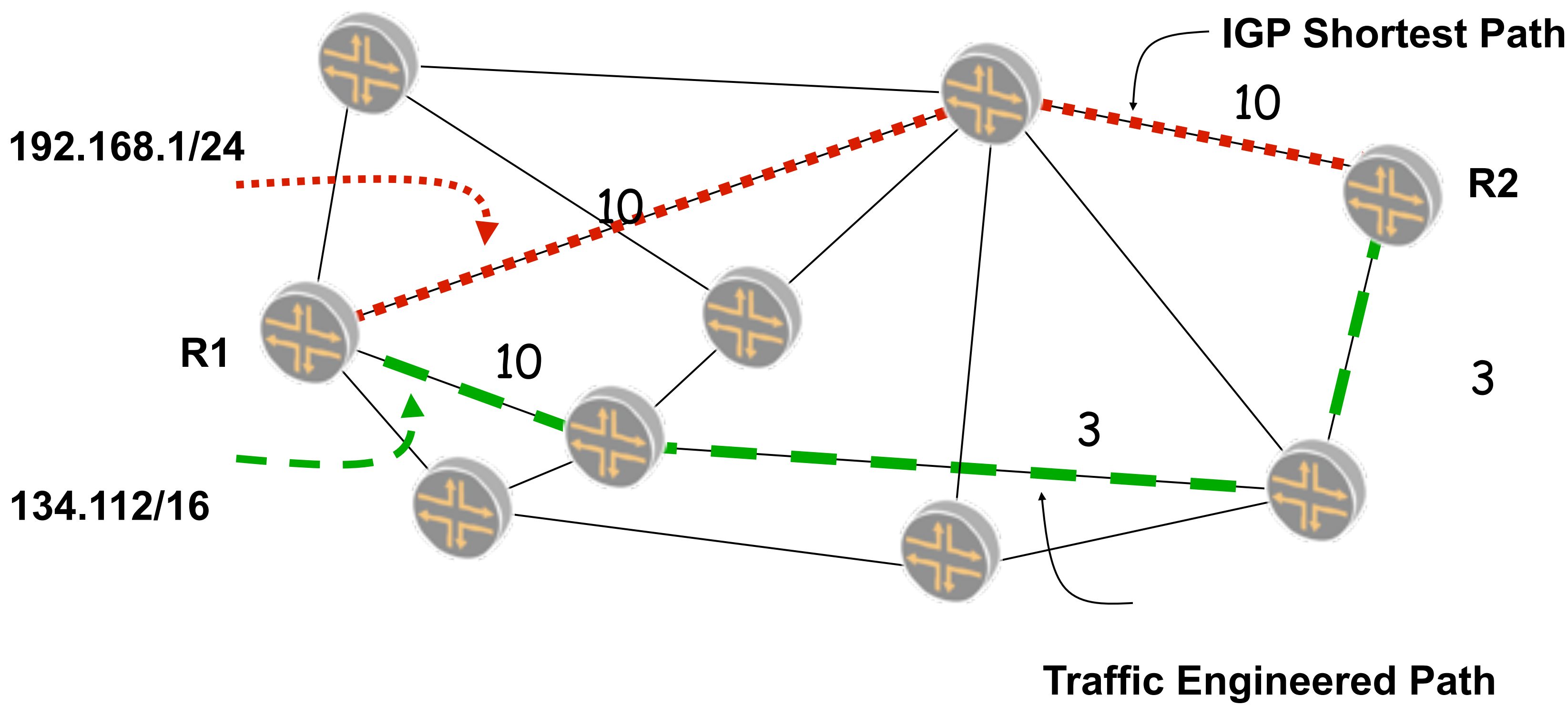
MPLS-Based Traffic Engineering



MPLS-Based Traffic Engineering



MPLS-Based Traffic Engineering



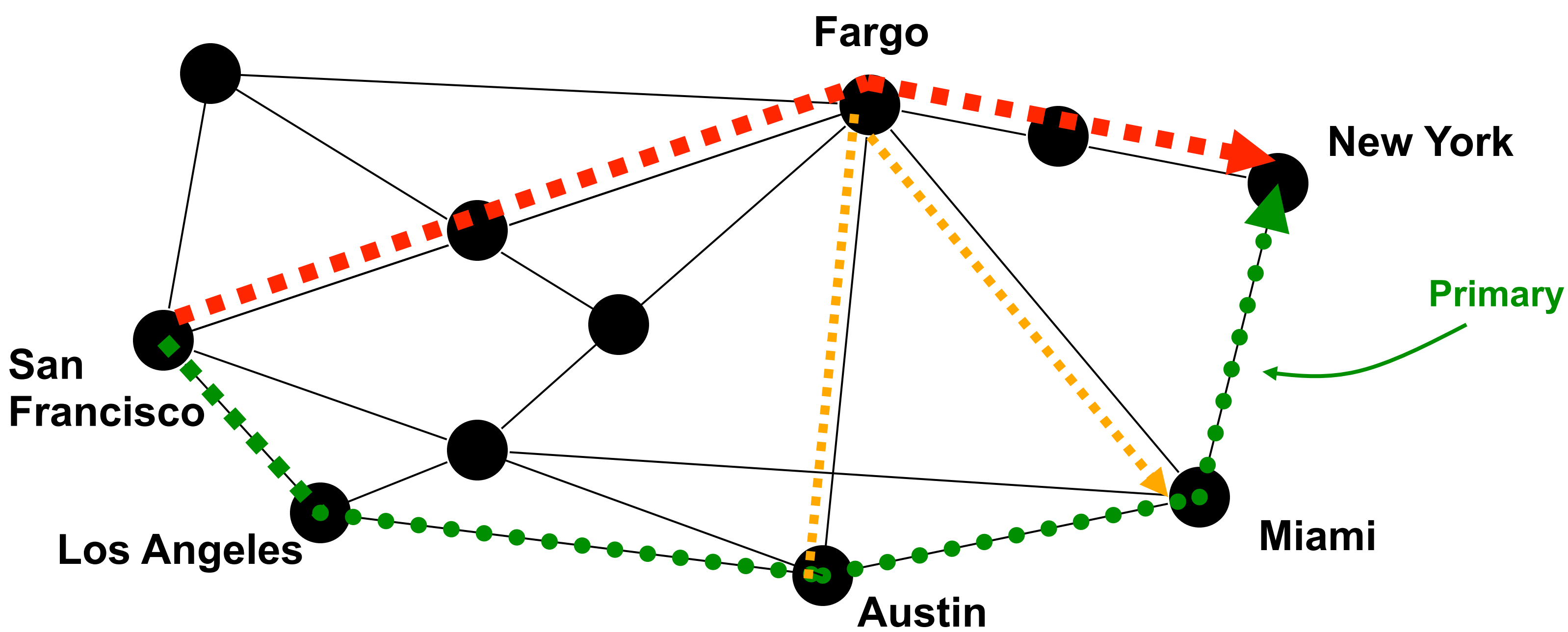
Traffic Engineering Options

- Can we do this another way
  - IGP metrics ☹
  - Flow = all traffic with same destination
- MPLS because
  - Granularity of flows
  - Flow = all traffic with same FEC
  - One network for all services
  - Less expensive

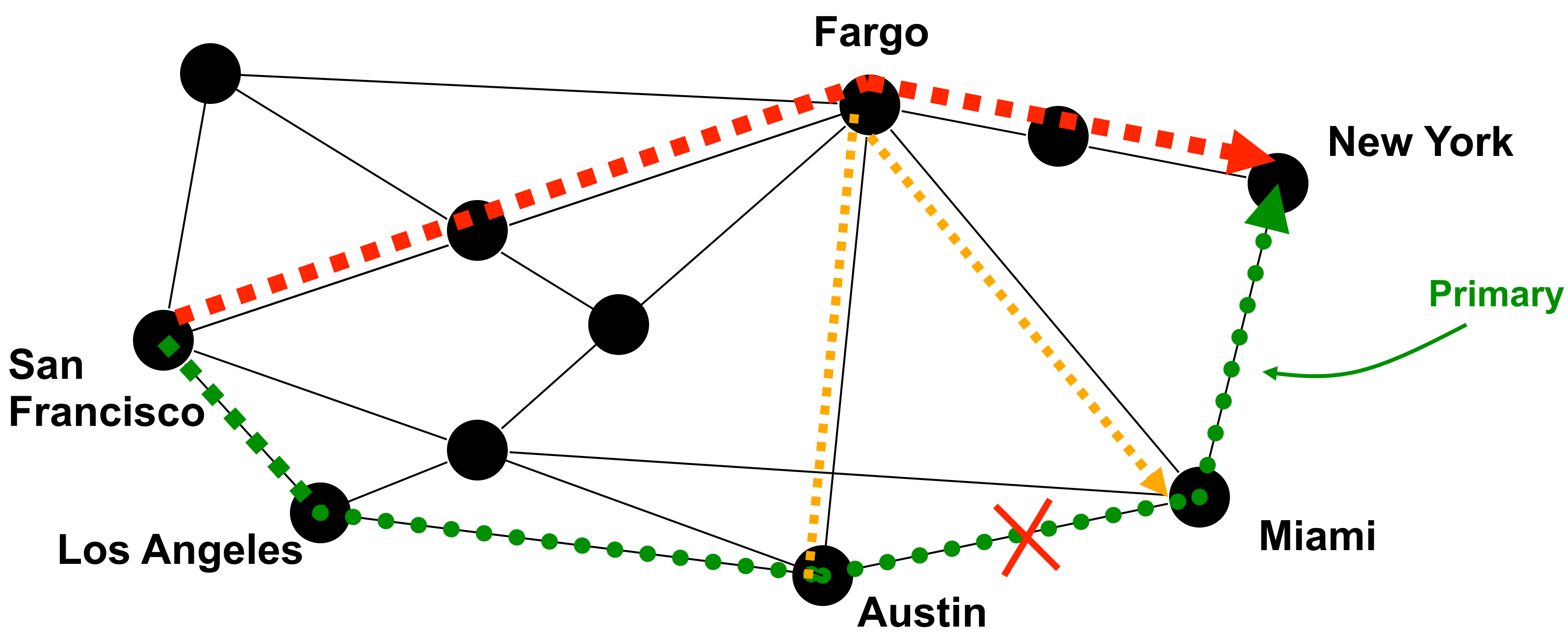
Traffic Protection

- Working definition
  - Reduce time of disruption
  - Reduce Packet Loss
  - "SONET like" sub millisecond recovery under failure conditions
- Can we do this another way
  - SONET/SDH
  - Lower IGP timers
- MPLS because
  - No extra capital – config change only
  - Pick which traffic needs it
  - One network for all services
  - Less expensive

Traffic Protection – example



Traffic Protection – example



Traffic Protection Variations

- Fast reroute
- Link Protection
- Link-Node Protection

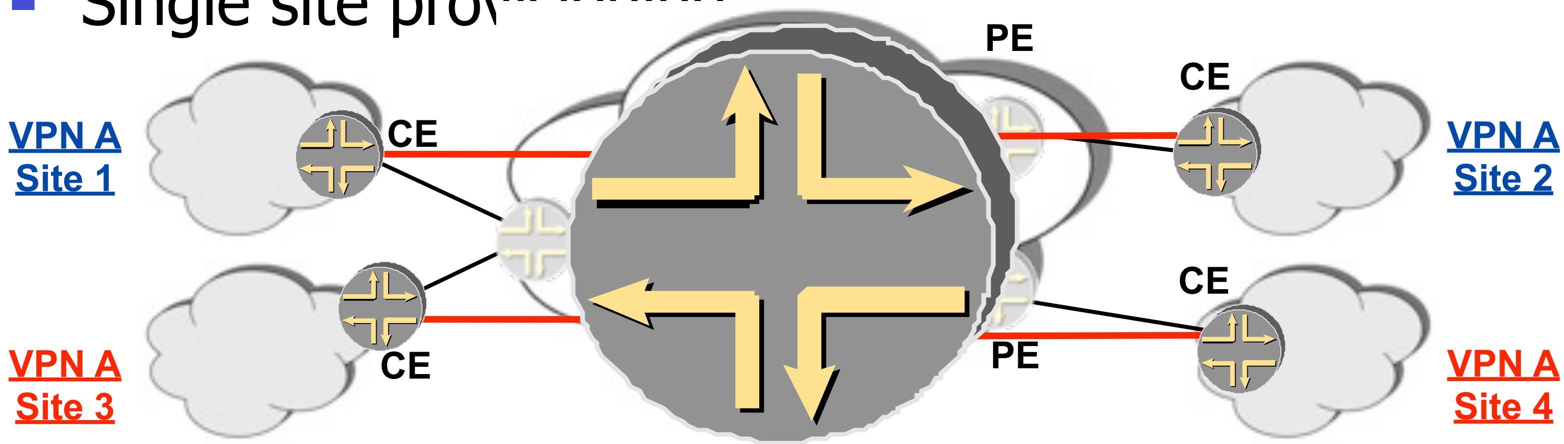
Layer 3 VPN (2547bis BGP/MPLS VPN)

Provider provisioned VPN

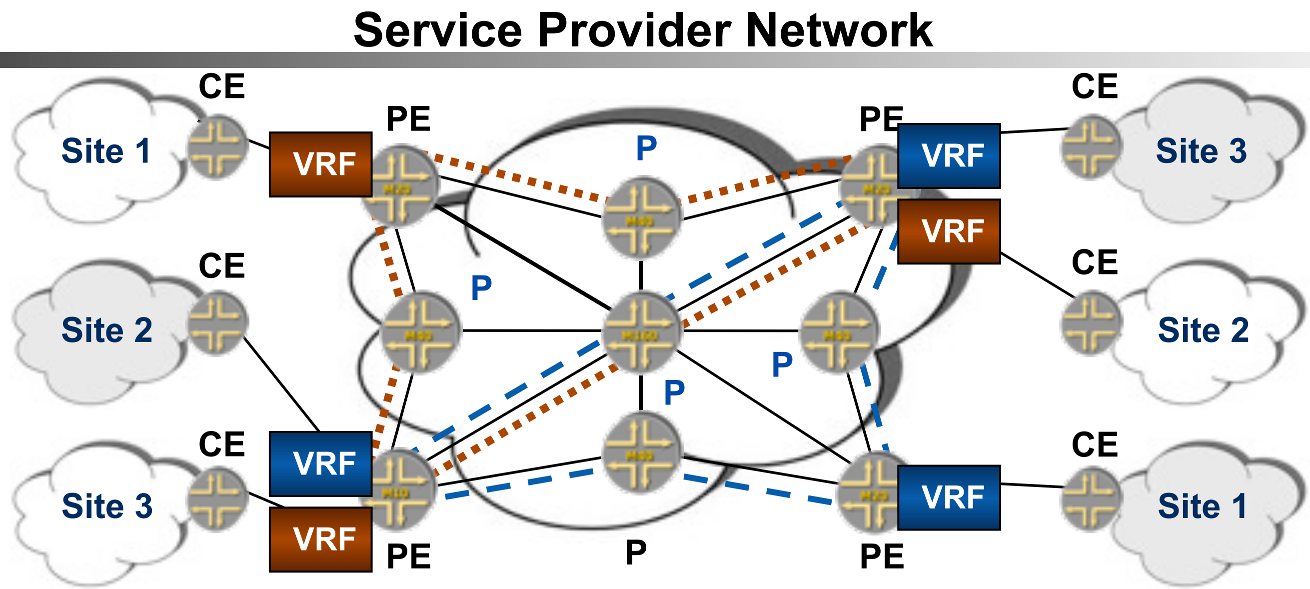
- ISP runs backbone for customer
  - Customer can be another ISP!
- Attractive to
  - Customer who do not want to run their own backbone
- Not attractive to
  - Customer who doesn't trust carrier
  - Customers who's jobs are threatened

Customer View of L3VPN

- Make the cloud look like a router
- Single site provisioning



Layer 3 PP-VPNs: RFC 2547bis (1 of 2)



- Application: Outsource VPN
  - PE router maintains VPN-specific forwarding tables for each of its directly connected VPNs
  - Conventional IP routing between CE and PE routers
  - VPN routes distributed using MP-BGP
    - Uses extended communities
  - VPN traffic forwarded across provider backbone using MPLS

Layer 3 PP-VPNs: RFC 2547bis (2 of 2)

- LDP or RSVP is used to set up PE-to-PE LSPs
- MP-BGP is used to distribute information about the VPN
  - Routing and reachability for the VPN
  - Labels for customer sites (tunneled in PE-PE LSP)
- Constrain connectivity by route filtering
  - Flexible, policy-based control mechanism

L3 VPN Options

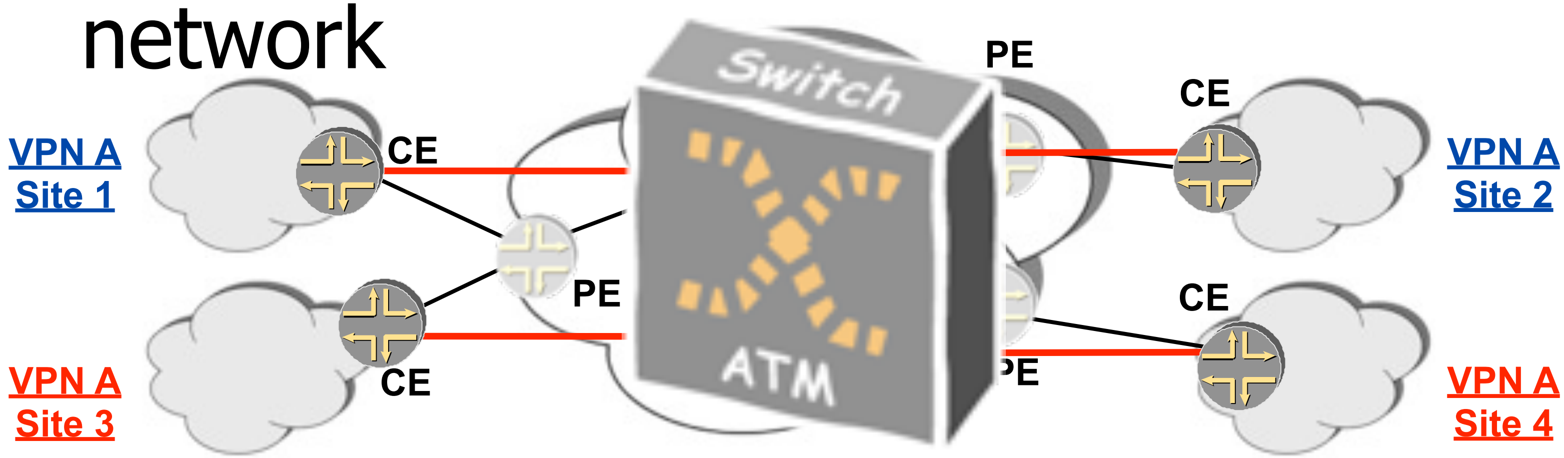
- Can we do it another way
  - Separate Physical routers
  - Separate Logical Routers
- MPLS because
  - Scaling
  - Single site provisioning
  - Less expensive

Layer 2 VPN's

- Provider provisioned VPN
  - ISP runs backbone for customer
    - Customer can be another ISP!
- Attractive to
  - Customers who want to preserve current CE technology
  - Customers who don't trust provider with L3
  - Carriers who want to offer another service
- Not Attractive to
  - Customers who do not want to run their own

Customer View of L2VPN

- Make the cloud look like a ATM/FR network



L2 VPN Options

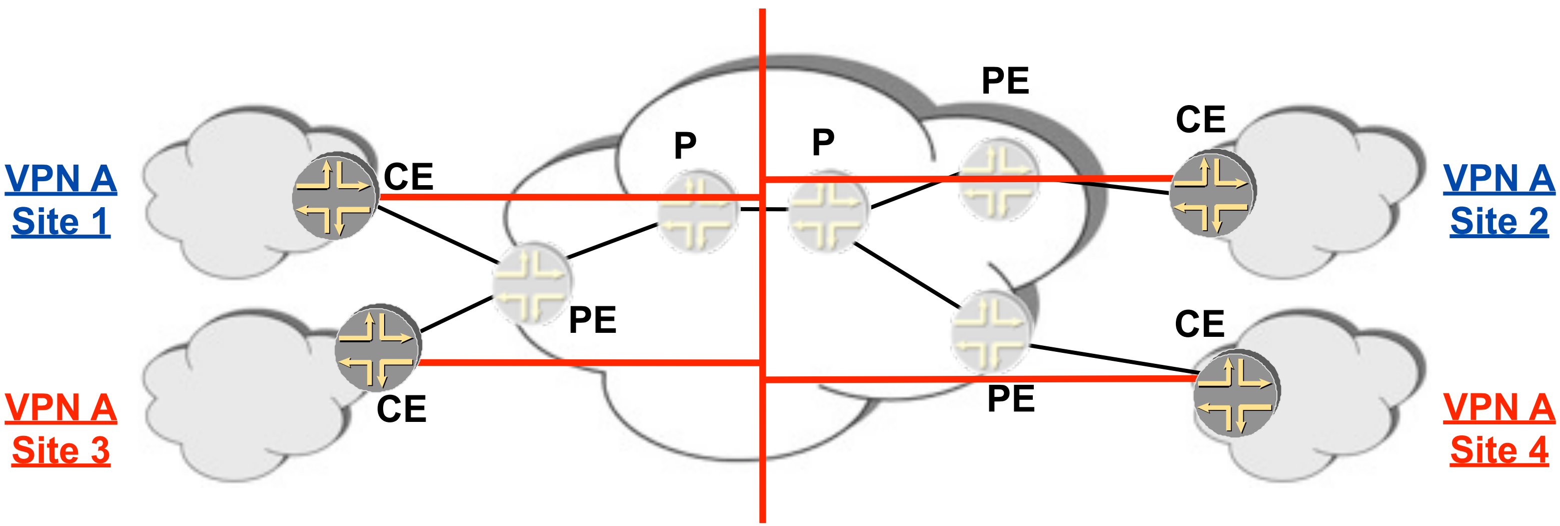
- Can we do it another way?
  - Traditional ATM/FR/leased line infrastructure
- MPLS because
  - One network for all services
  - Less expensive
  - Scaling
  - Single site provisioning *

VPLS

- Virtual Private LAN Service
- Attractive to
  - Customers who like ethernet as CE
  - Lots of locations close together with 'high' WAN bandwidth requirements (kiosks)
  - No routing required
- Not attractive to
  - Customers who like control and visibility of core. "what can I ping to identify fault-domain?"
  - Controlling broadcasts

VPLS

- Make the cloud look like an ethernet switch

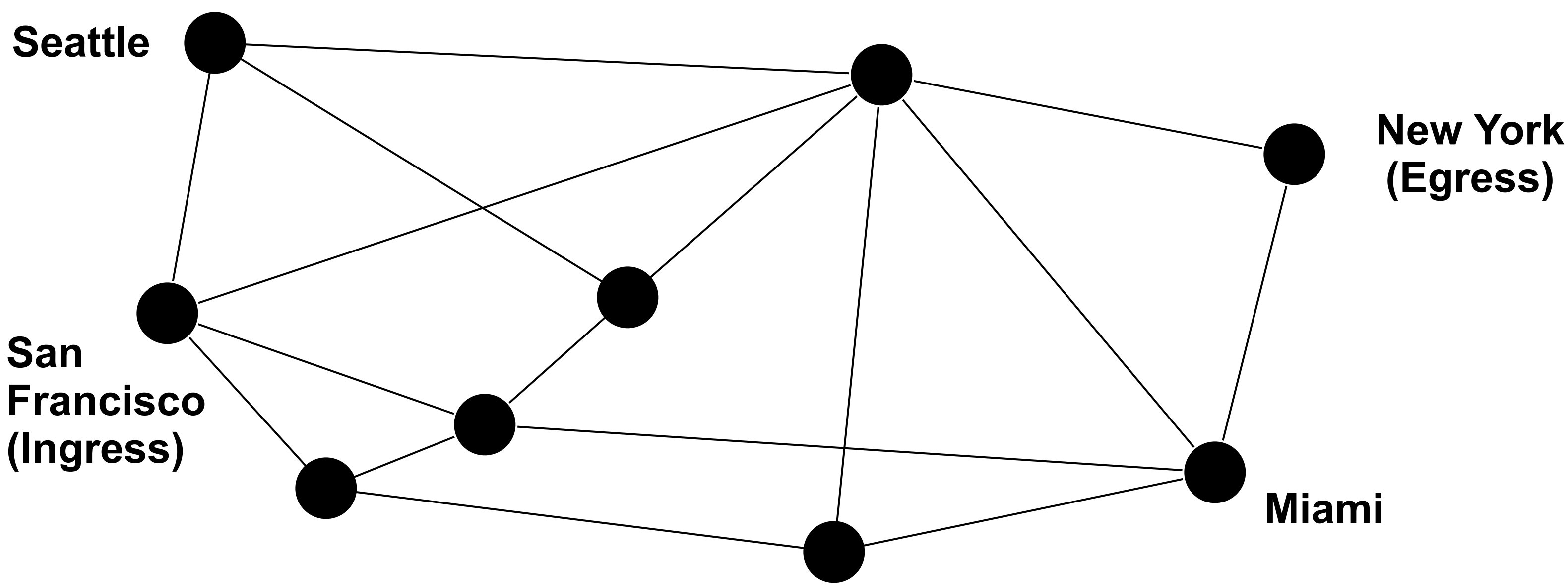


VPLS Options

- Can we do it another way?
  - Separate physical switches tying all customer sites
  - VLAN's over layer 2 backbone
- MPLS because
  - Scaling
  - One network for all services
  - Less expensive

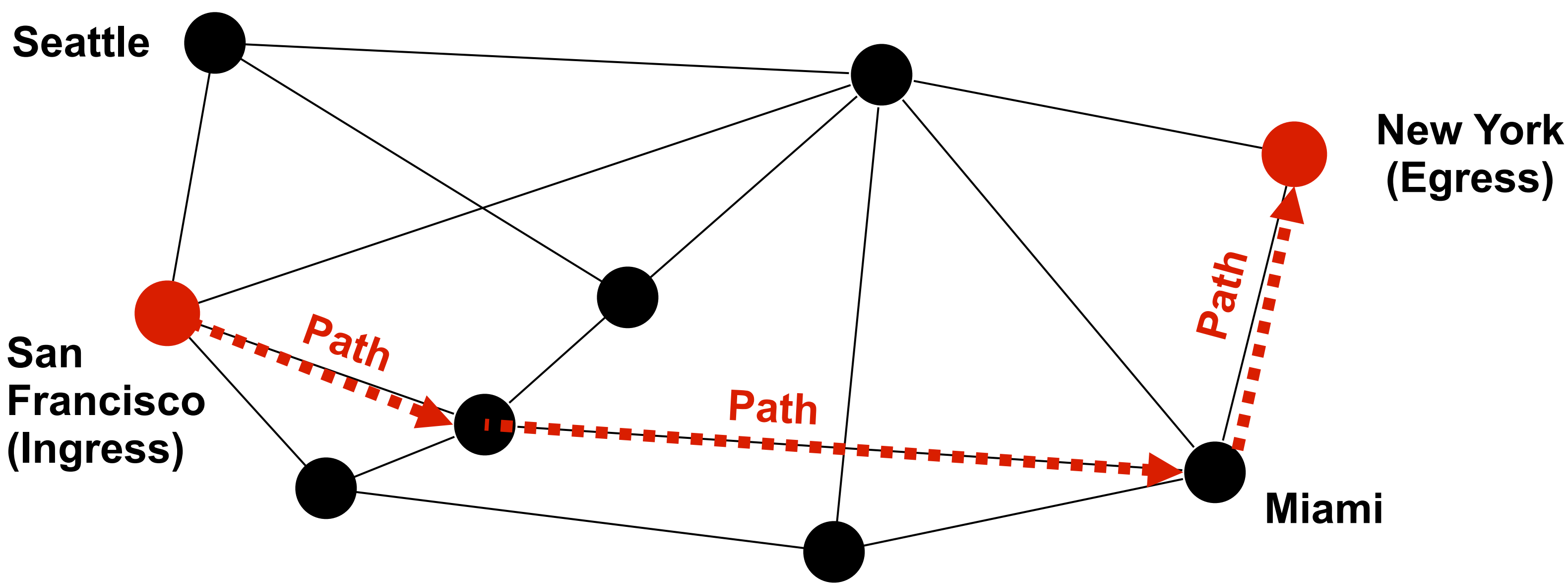
RSVP Signaling Example: Path

RSVP sets up path from San Francisco to New York



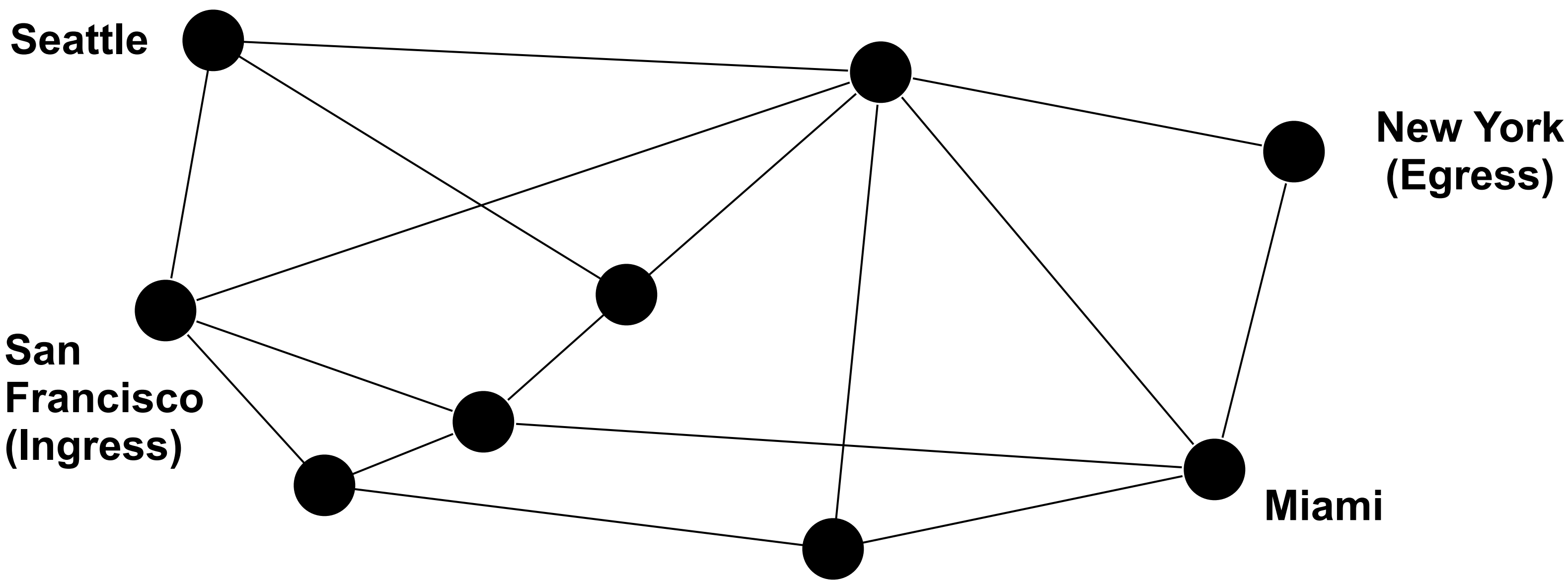
RSVP Signaling Example: Path

RSVP sets up path from San Francisco to New York



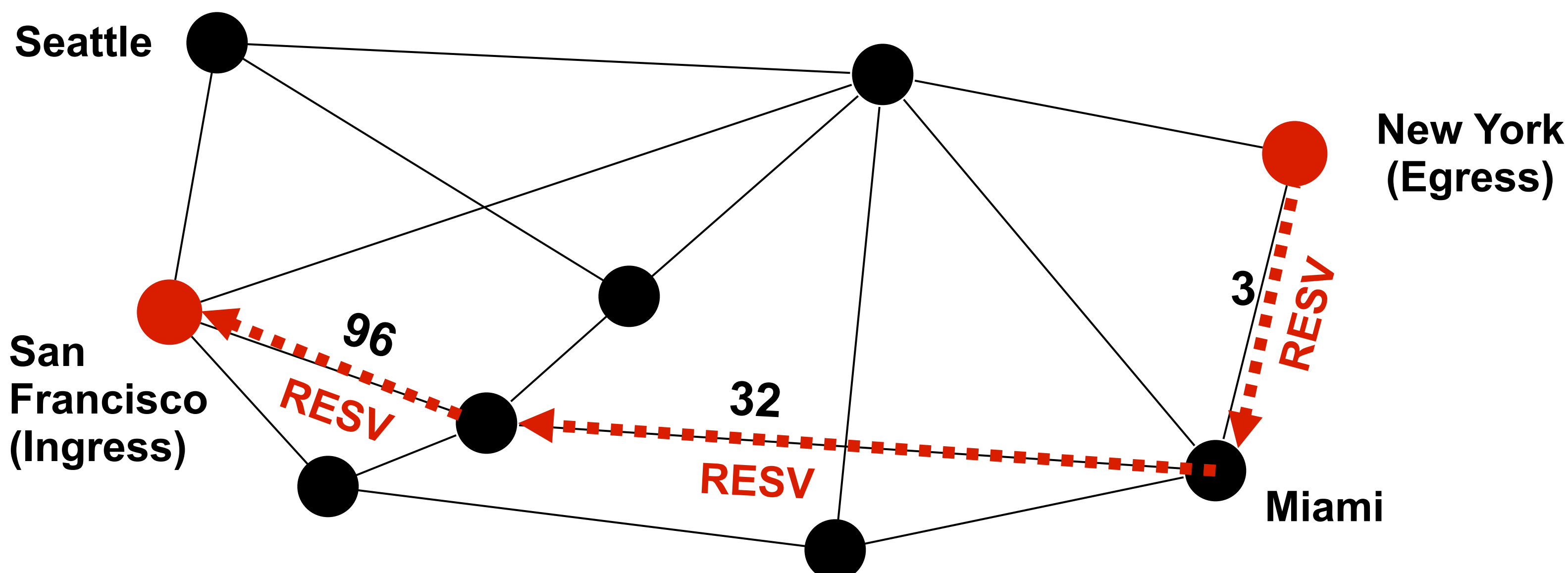
RSVP Signaling Example: Reservation

- The resv message visits each router on the path in reverse order
  - Labels assigned hop to hop in the upstream direction



RSVP Signaling Example: Reservation

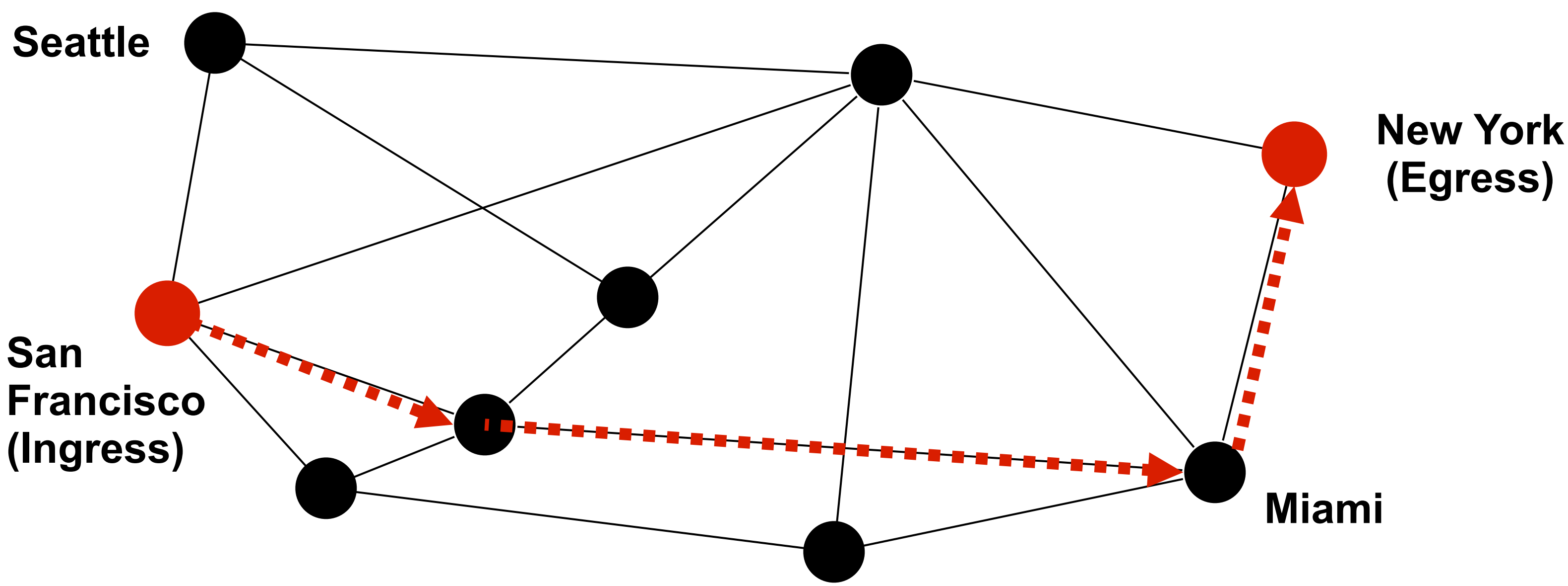
- The resv message visits each router on the path in reverse order
  - Labels assigned hop to hop in the upstream direction



LSP Established!

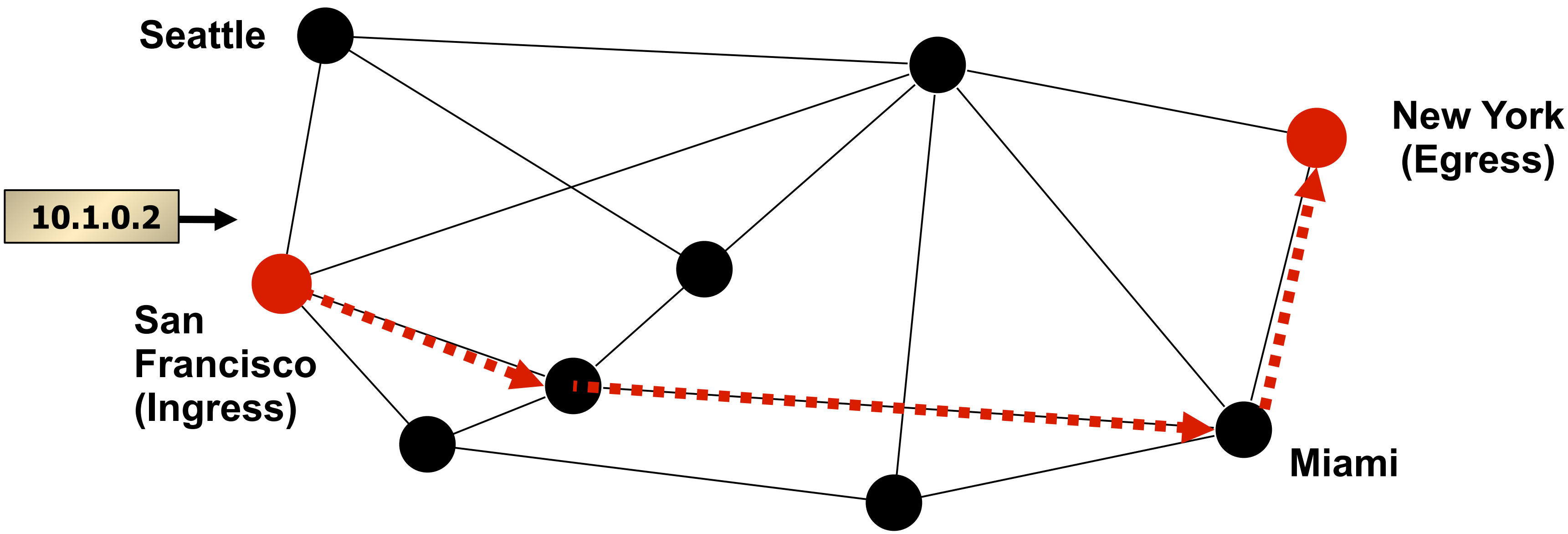
RSVP Signaling Example: Forwarding

RSVP sets up path from San Francisco to New York



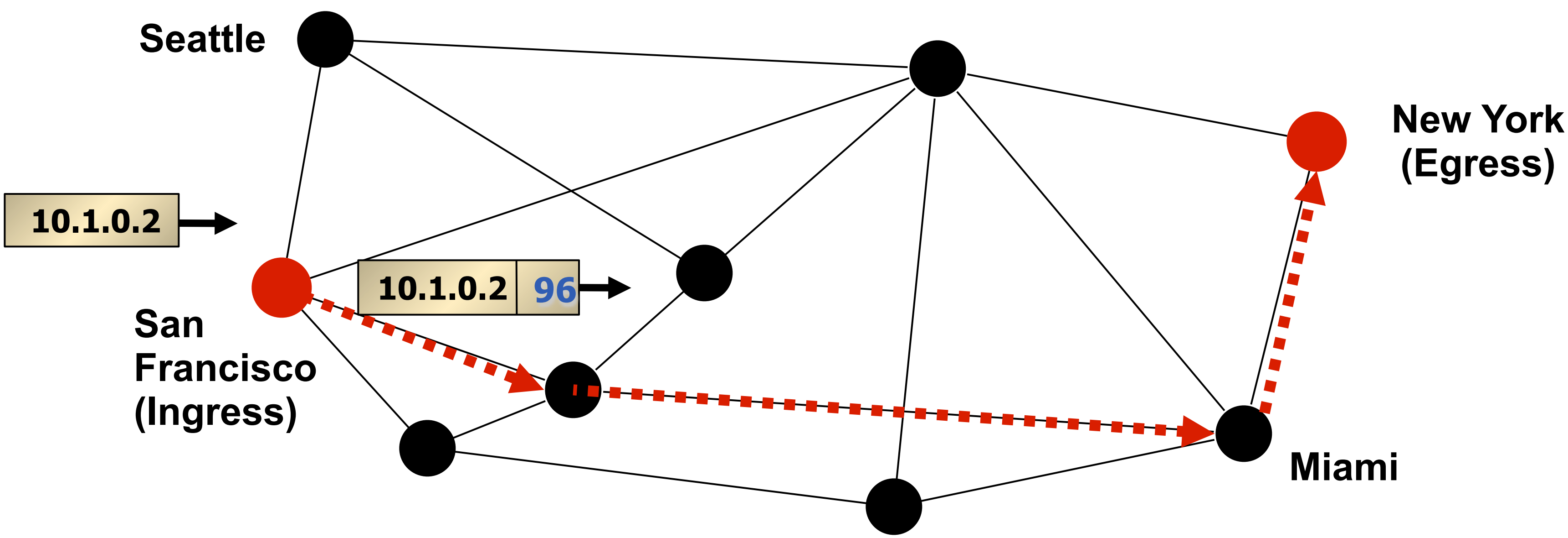
RSVP Signaling Example: Forwarding

RSVP sets up path from San Francisco to New York



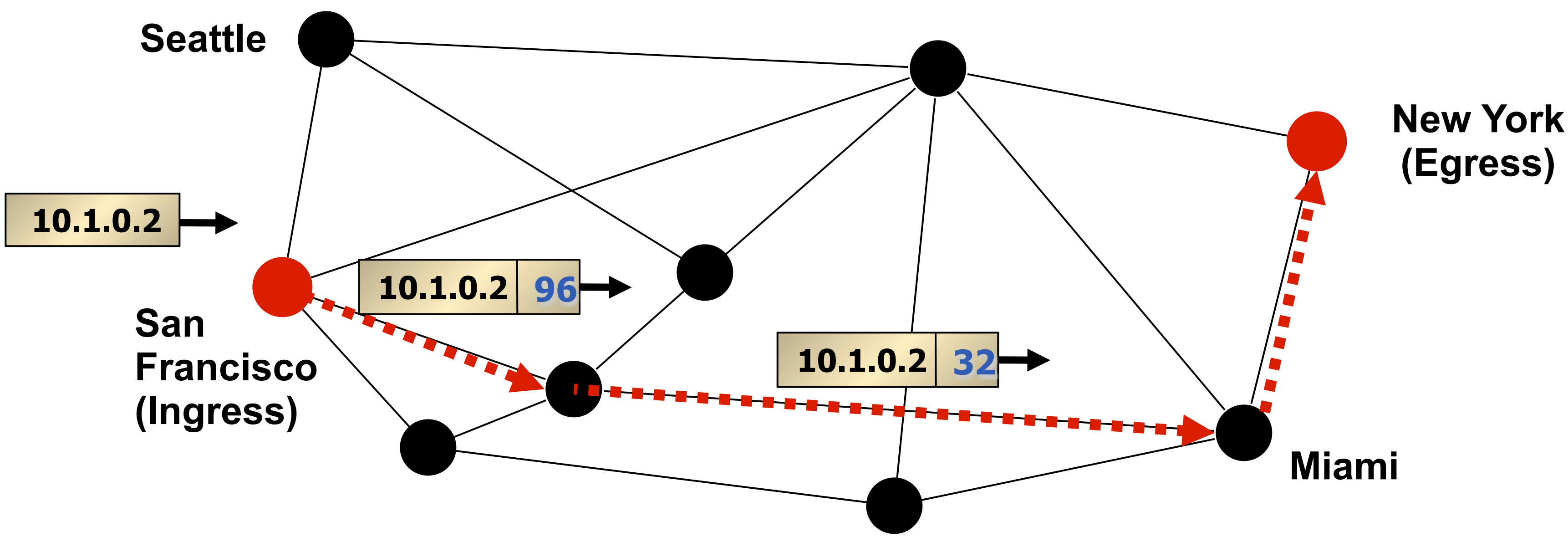
RSVP Signaling Example: Forwarding

RSVP sets up path from San Francisco to New York



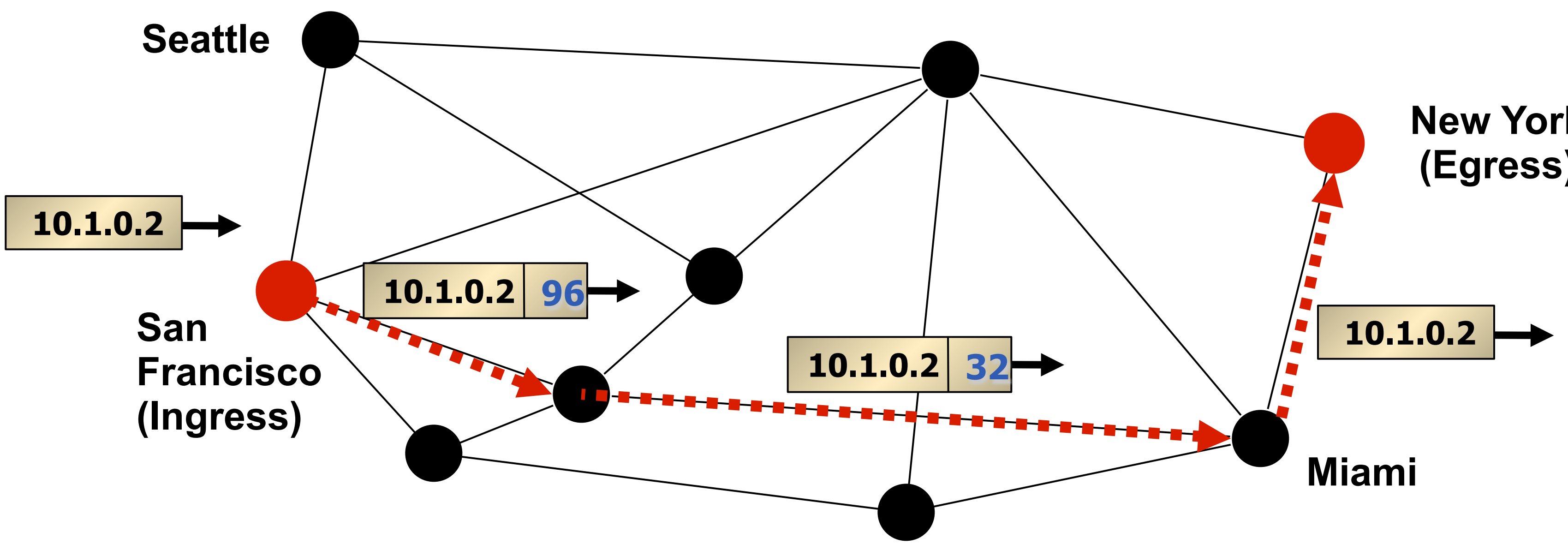
RSVP Signaling Example: Forwarding

RSVP sets up path from San Francisco to New York



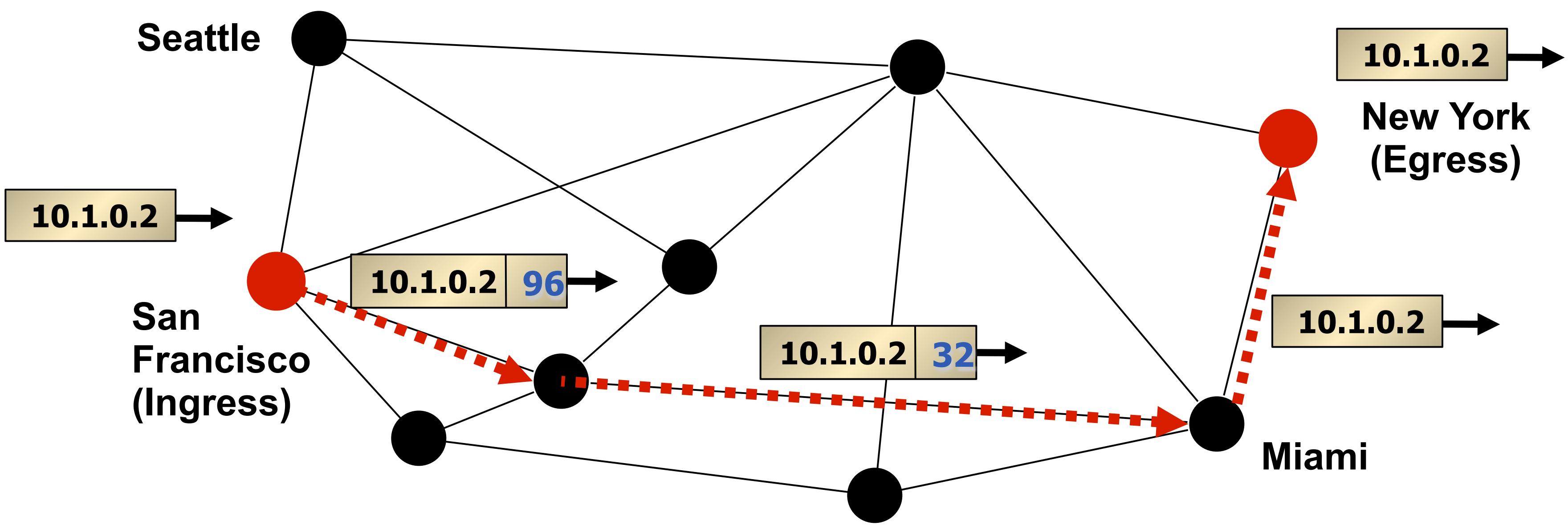
RSVP Signaling Example: Forwarding

RSVP sets up path from San Francisco to New York



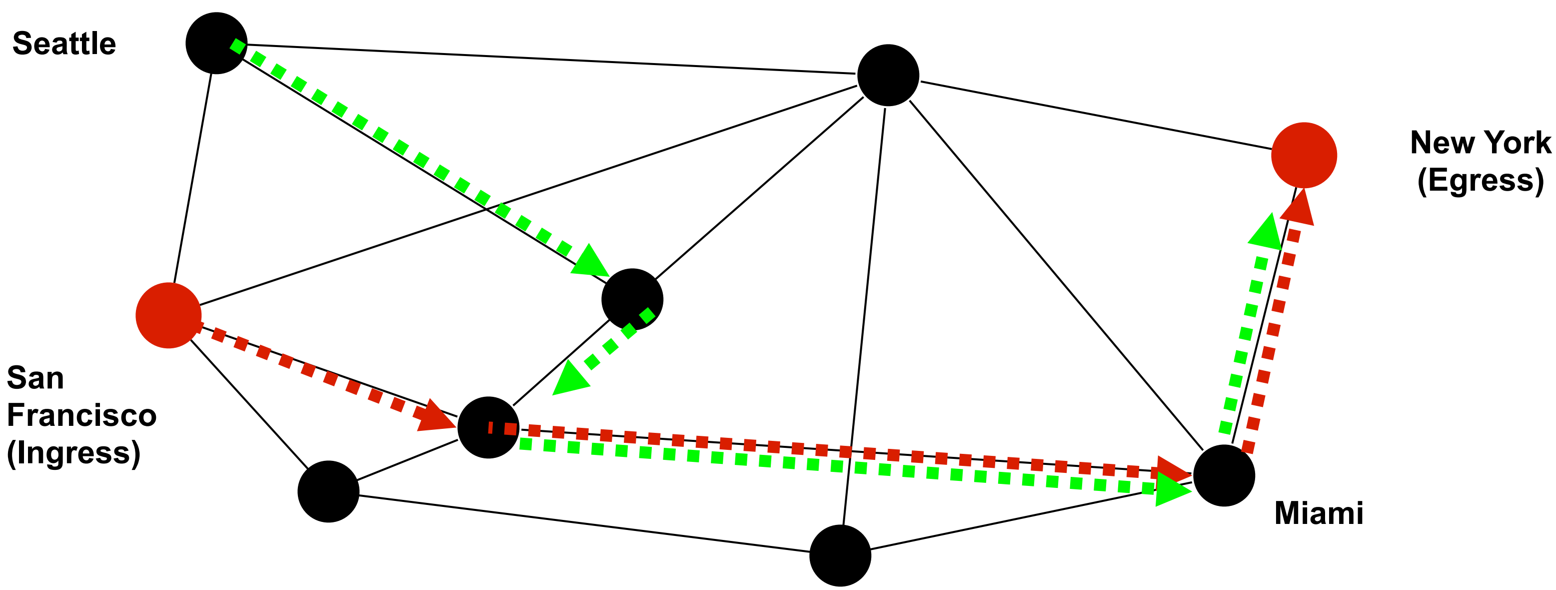
RSVP Signaling Example: Forwarding

RSVP sets up path from San Francisco to New York



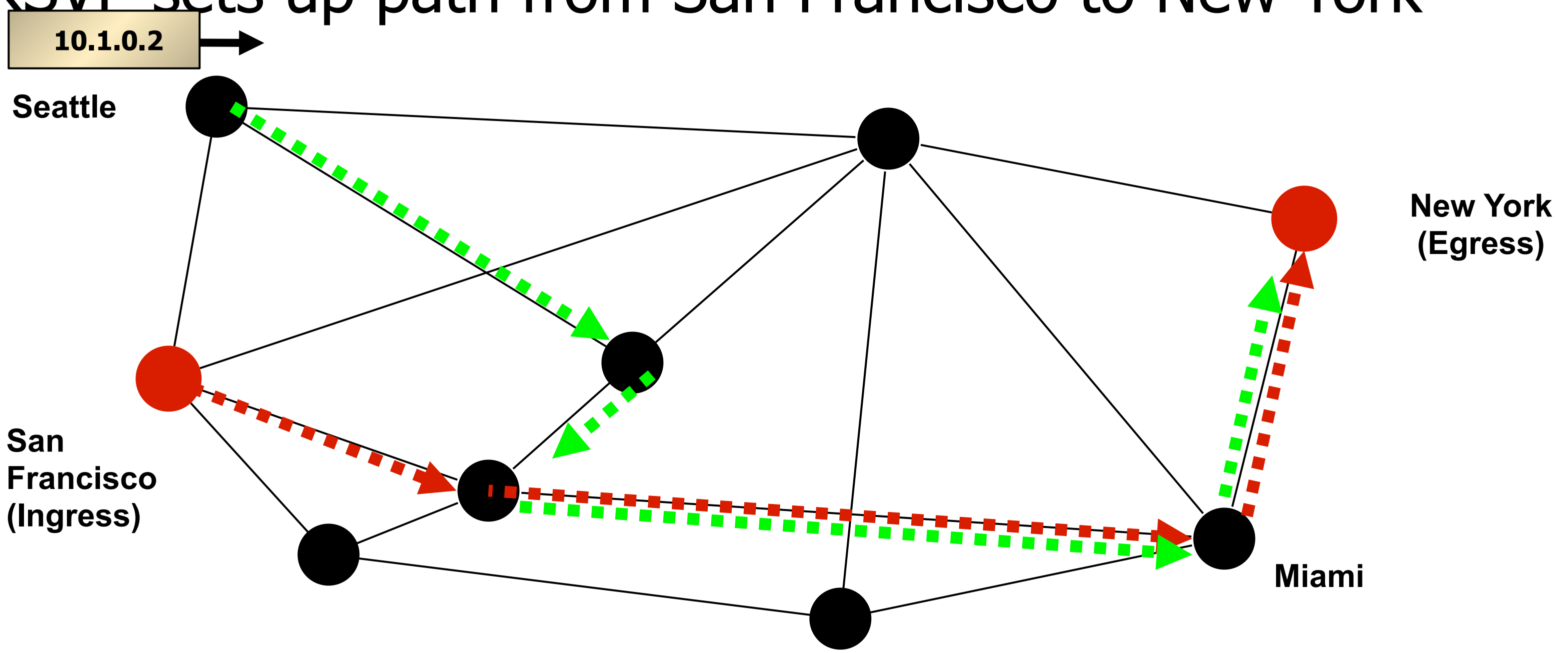
RSVP Signaling Example: Forwarding 2

RSVP sets up path from San Francisco to New York



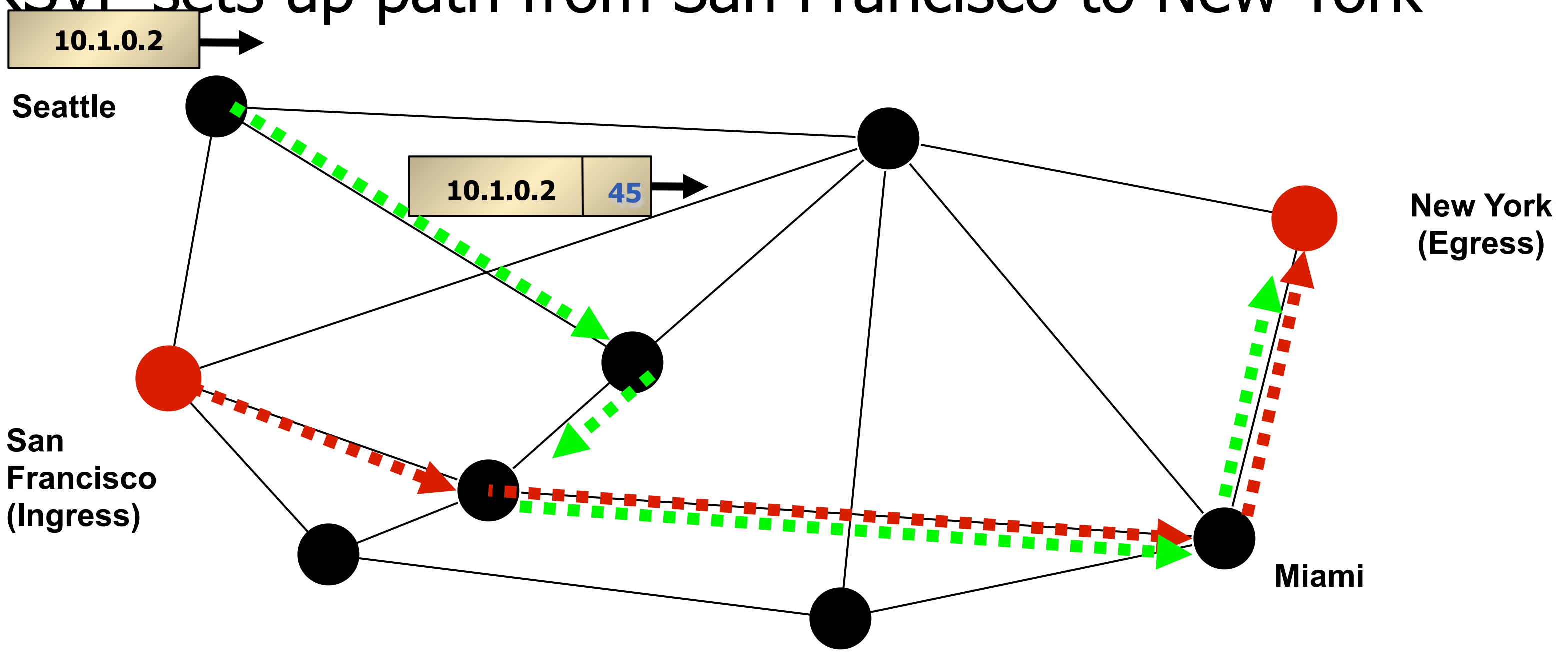
RSVP Signaling Example: Forwarding 2

RSVP sets up path from San Francisco to New York



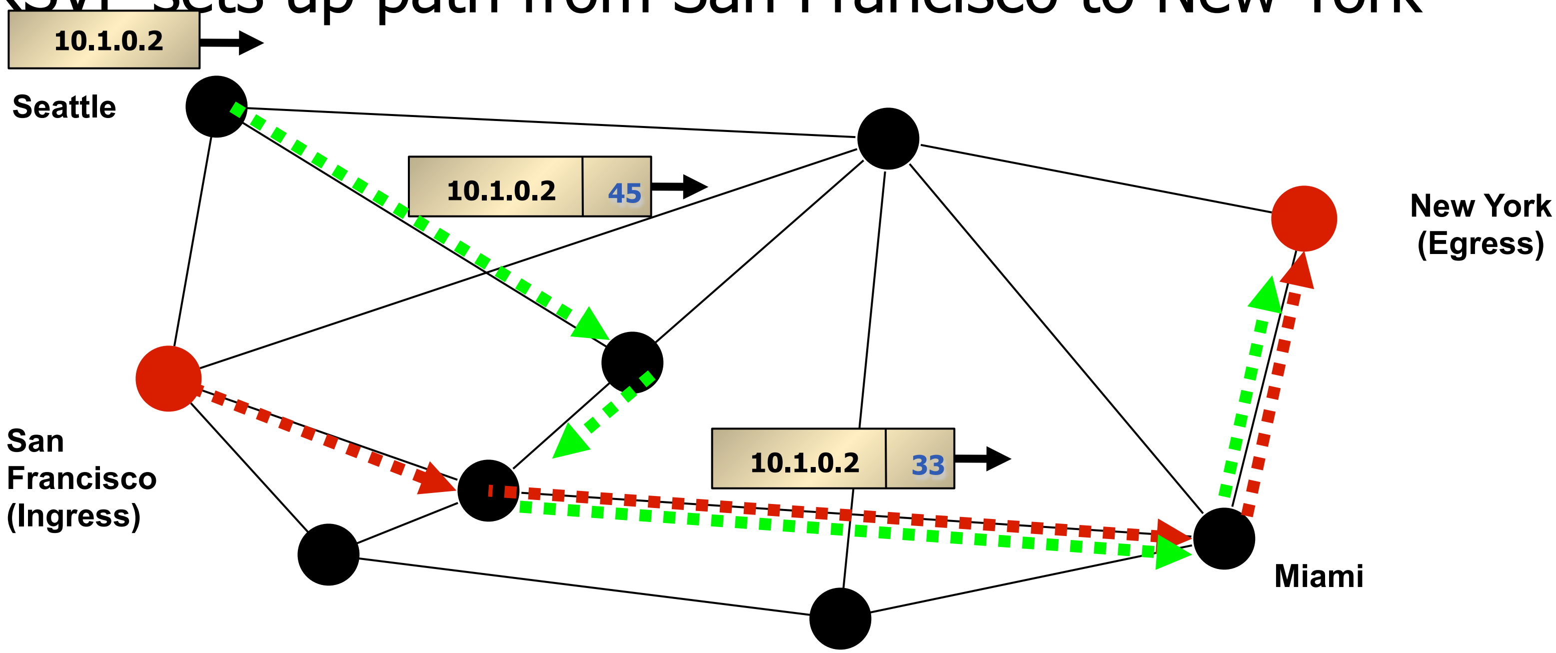
RSVP Signaling Example: Forwarding 2

RSVP sets up path from San Francisco to New York



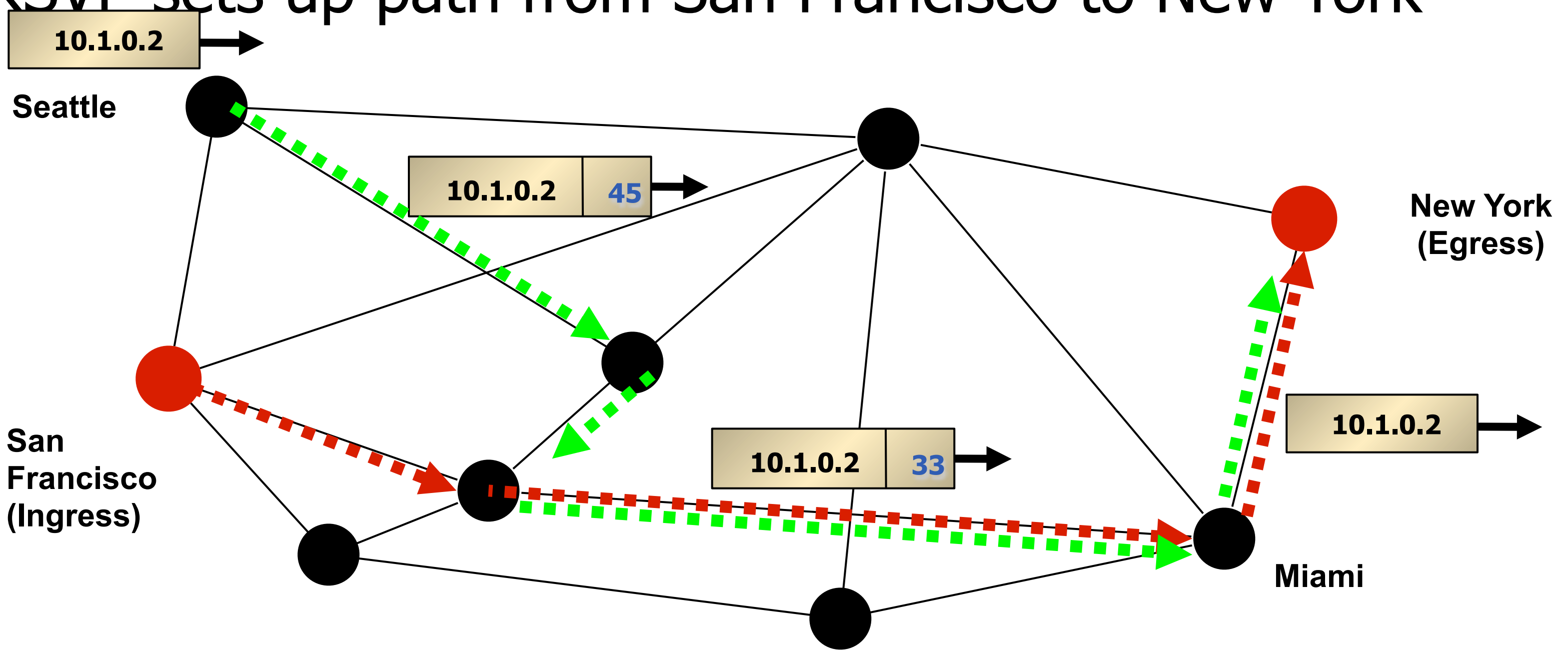
RSVP Signaling Example: Forwarding 2

RSVP sets up path from San Francisco to New York



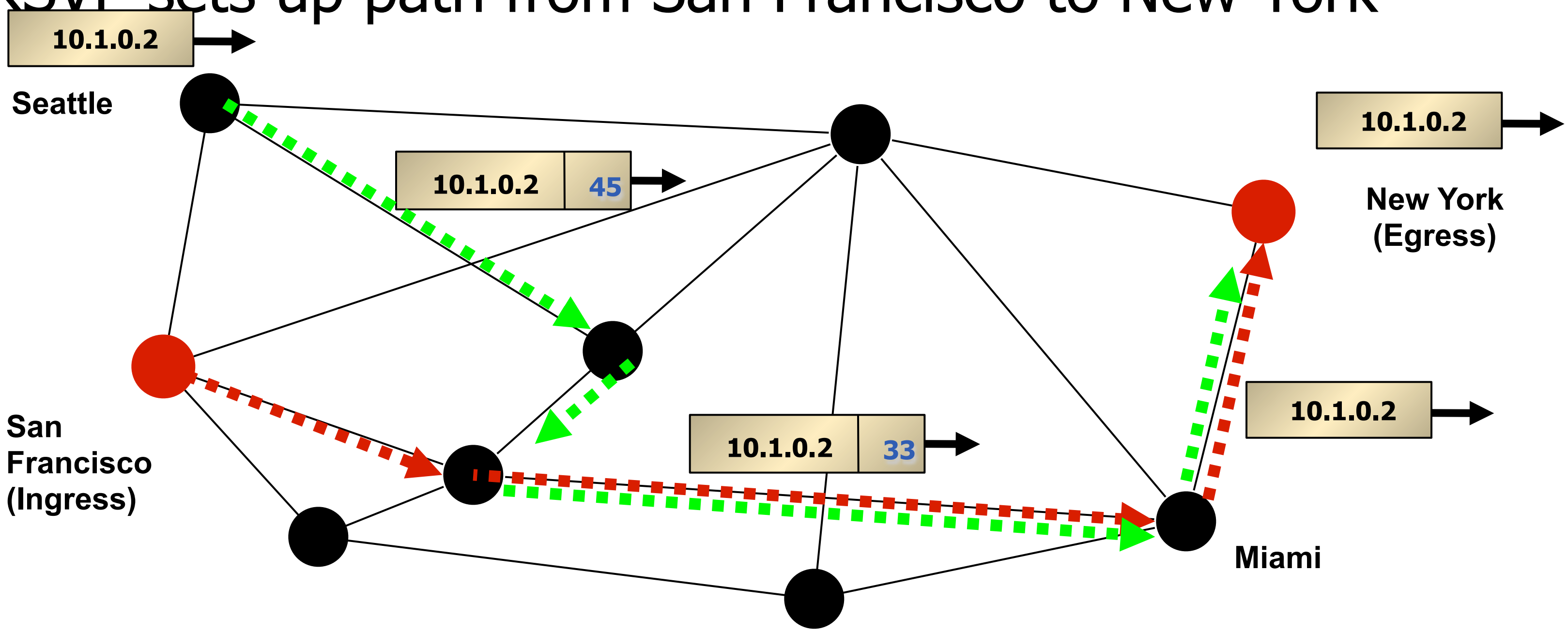
RSVP Signaling Example: Forwarding 2

RSVP sets up path from San Francisco to New York



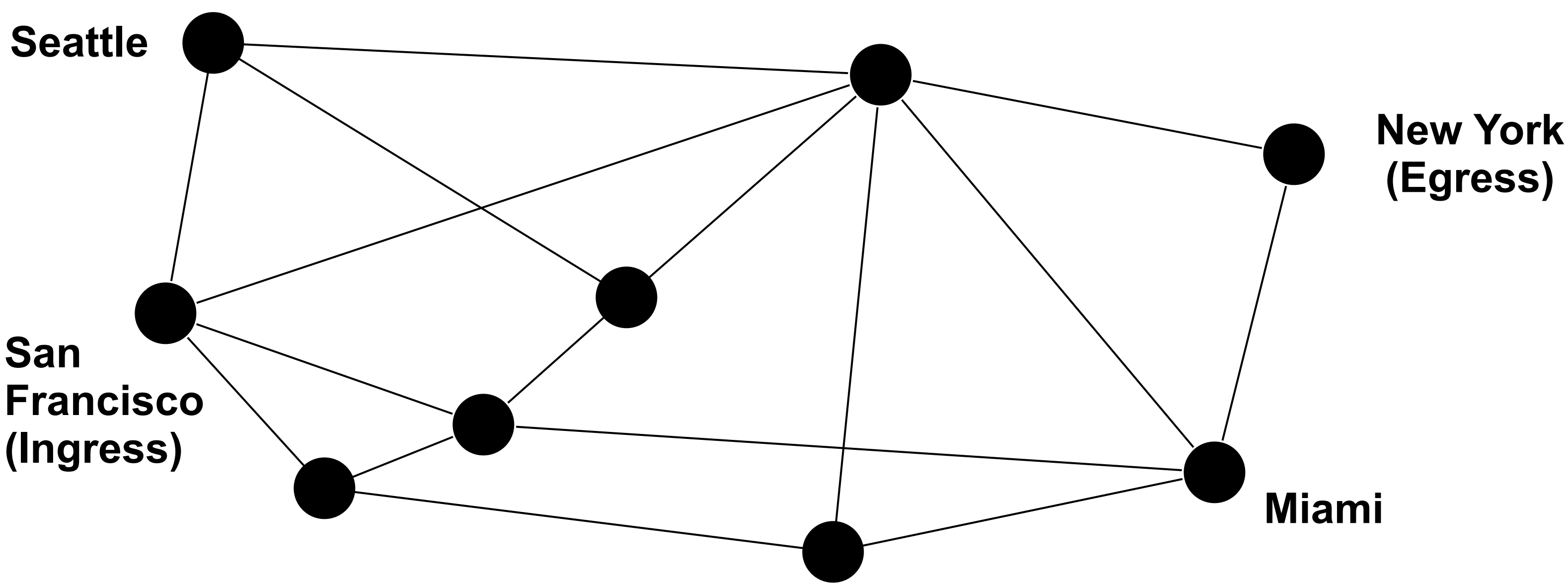
RSVP Signaling Example: Forwarding 2

RSVP sets up path from San Francisco to New York



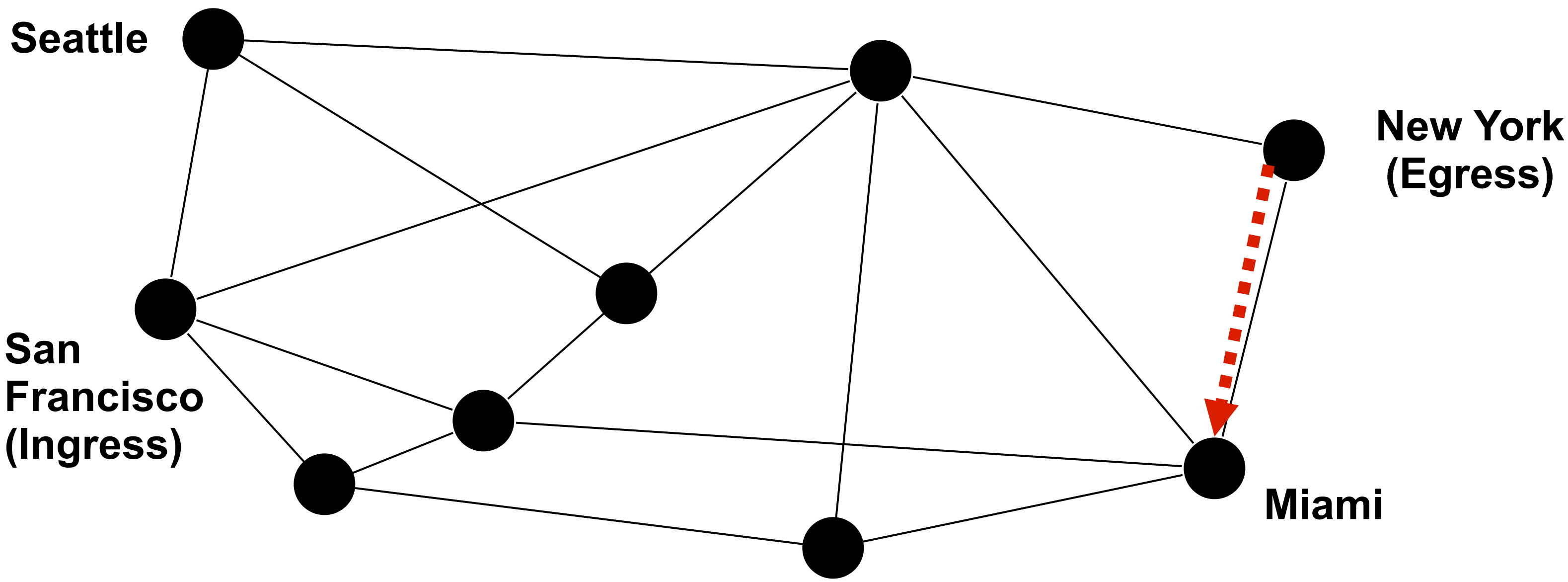
LDP Signaling Example: Label Binding

- Label Mappings are made for entries in the routing table
  - Labels assigned hop to hop in the upstream direction



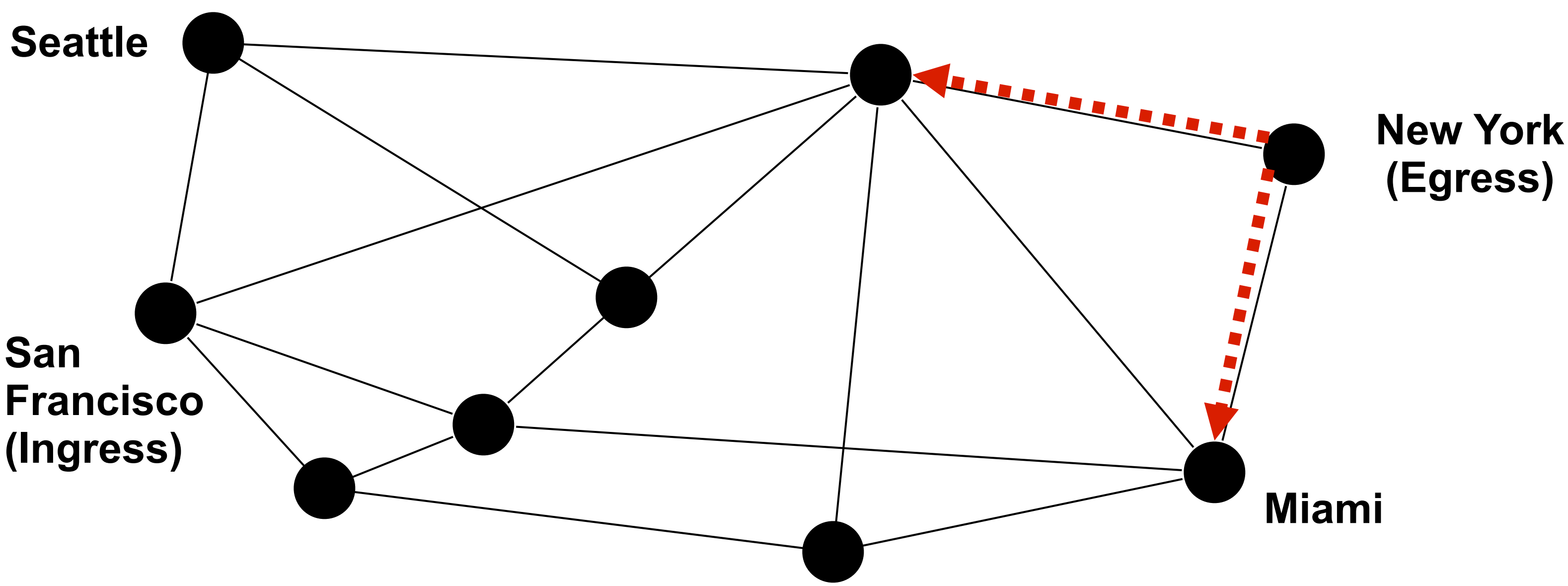
LDP Signaling Example: Label Binding

- Label Mappings are made for entries in the routing table
  - Labels assigned hop to hop in the upstream direction



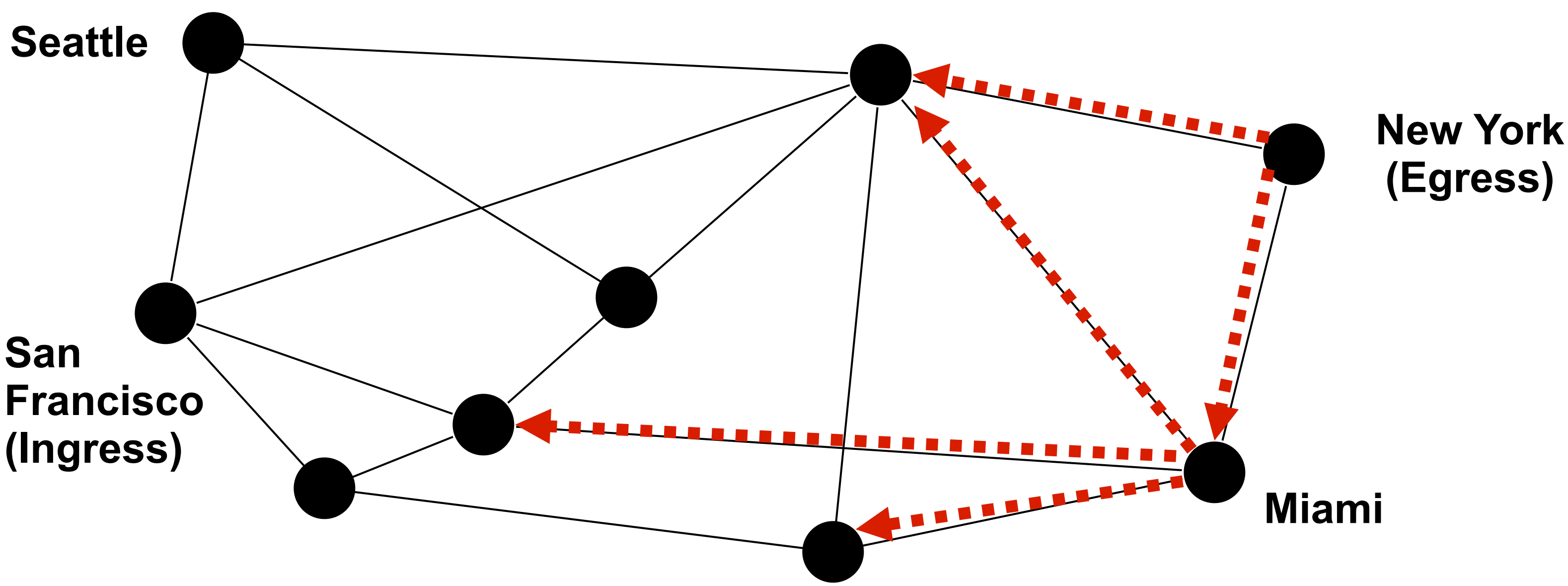
LDP Signaling Example: Label Binding

- Label Mappings are made for entries in the routing table
  - Labels assigned hop to hop in the upstream direction



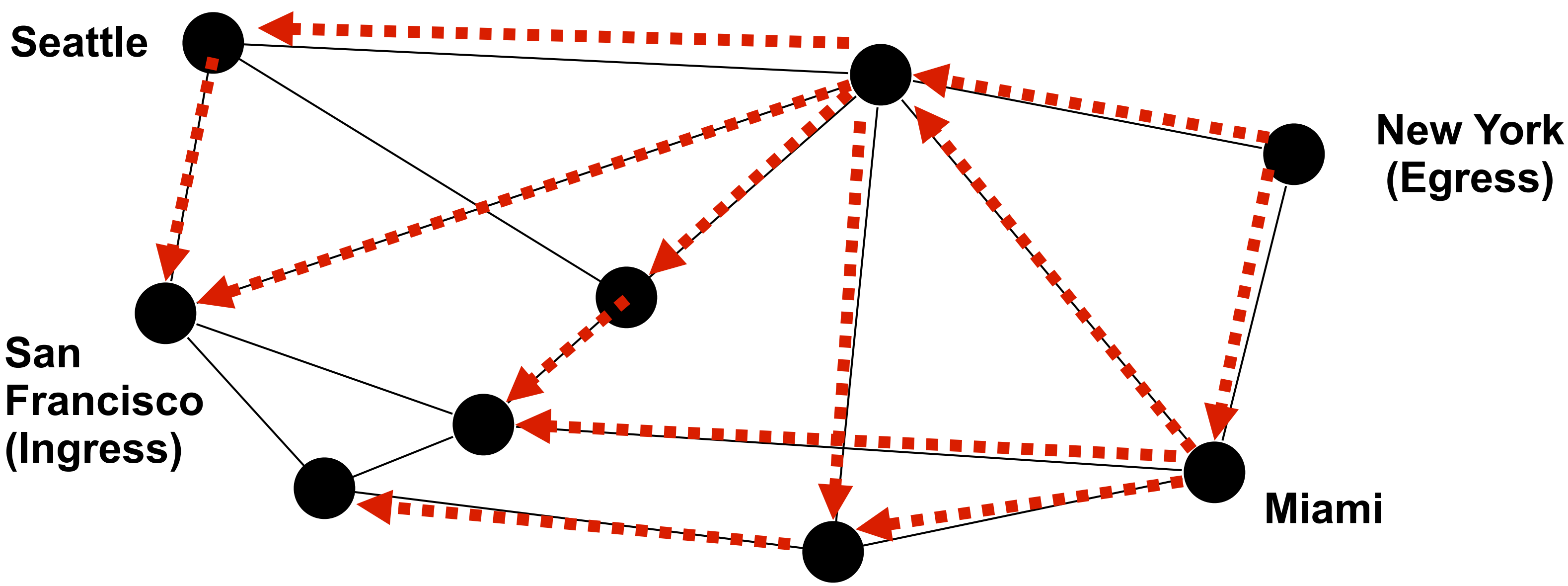
LDP Signaling Example: Label Binding

- Label Mappings are made for entries in the routing table
  - Labels assigned hop to hop in the upstream direction



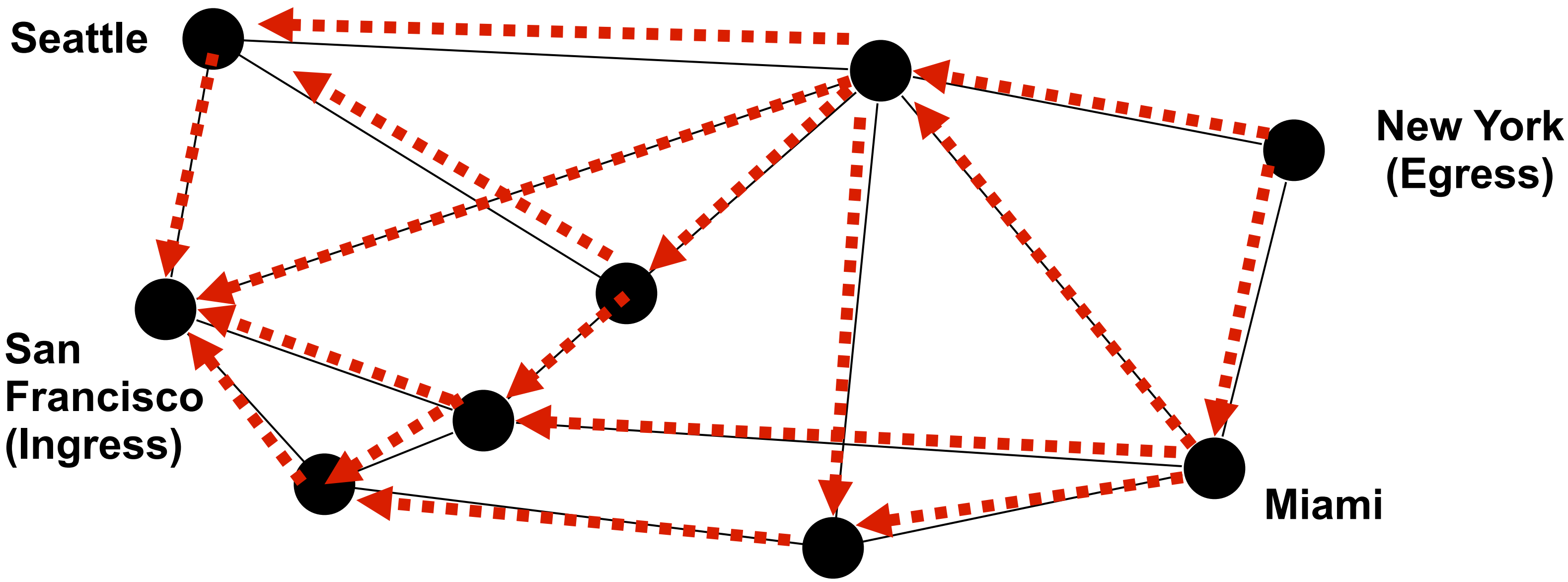
LDP Signaling Example: Label Binding

- Label Mappings are made for entries in the routing table
  - Labels assigned hop to hop in the upstream direction



LDP Signaling Example: Label Binding

- Label Mappings are made for entries in the routing table
  - Labels assigned hop to hop in the upstream direction

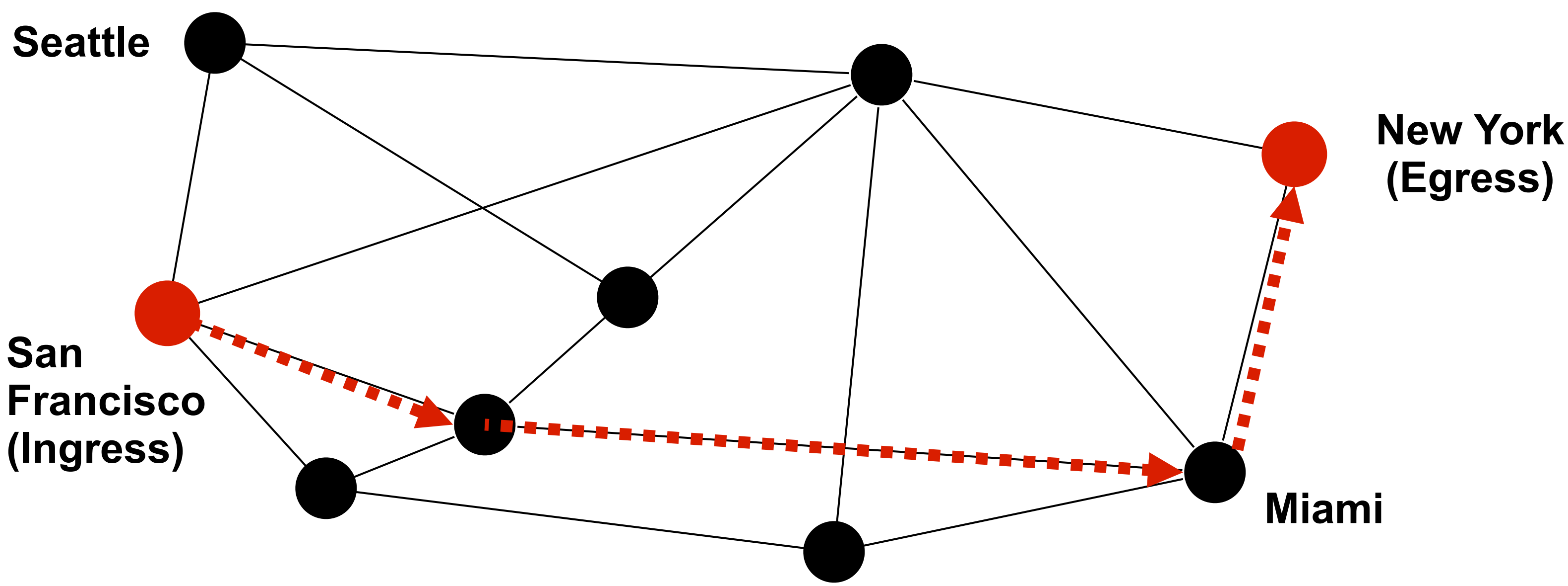


For those who care!

- The last slide assumed LDP was operating in
  - Unsolicited Downstream mode
    - Not downstream-on-demand
  - Ordered Mode
    - Not Independent Mode
  - Liberal label retention
    - Not conservative

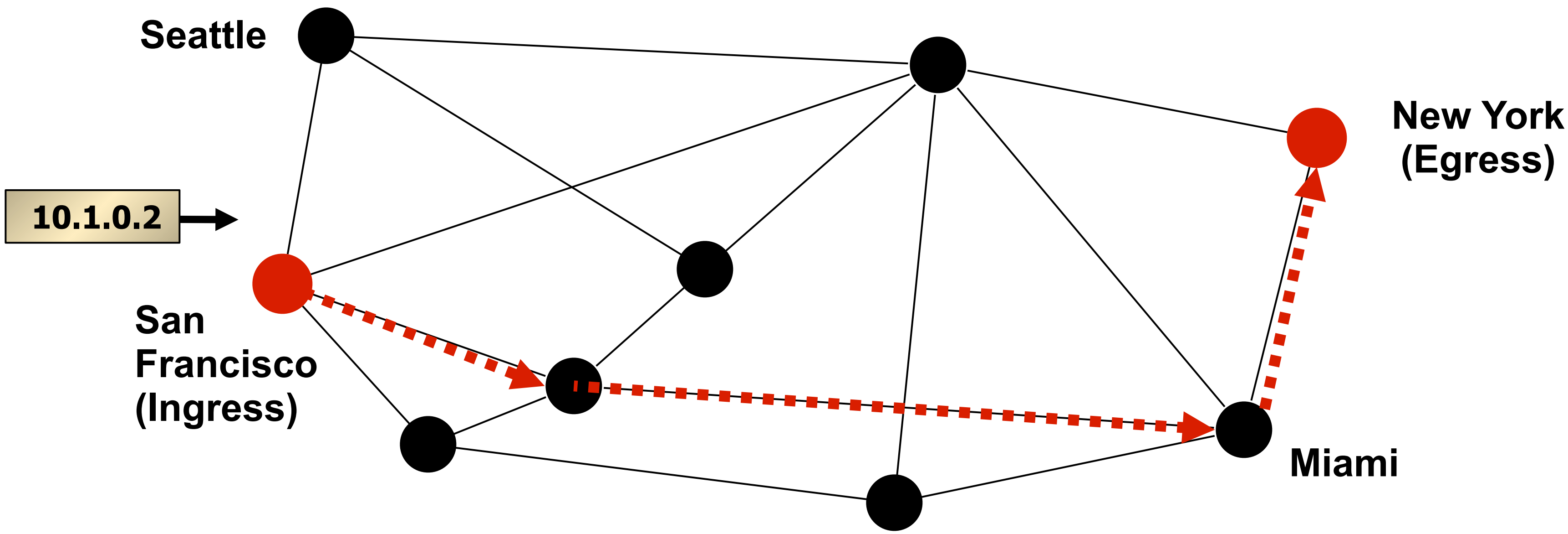
LDP Signaling Example: Forwarding

LDP path available to egress



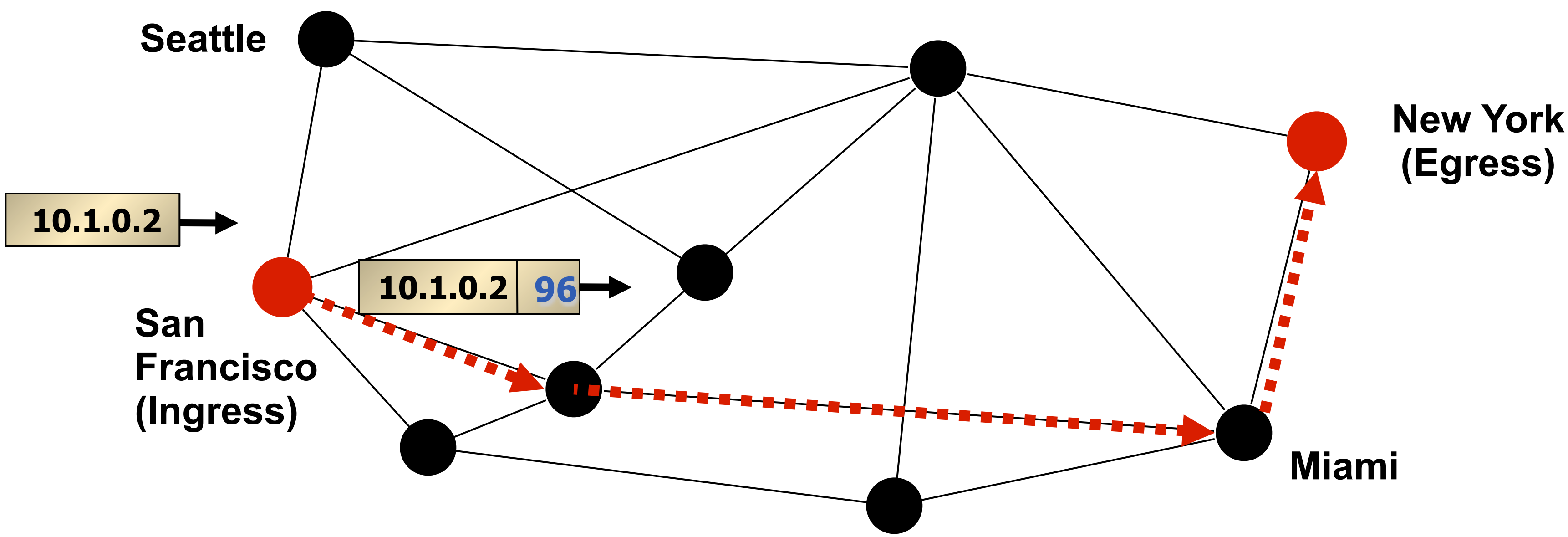
LDP Signaling Example: Forwarding

LDP path available to egress



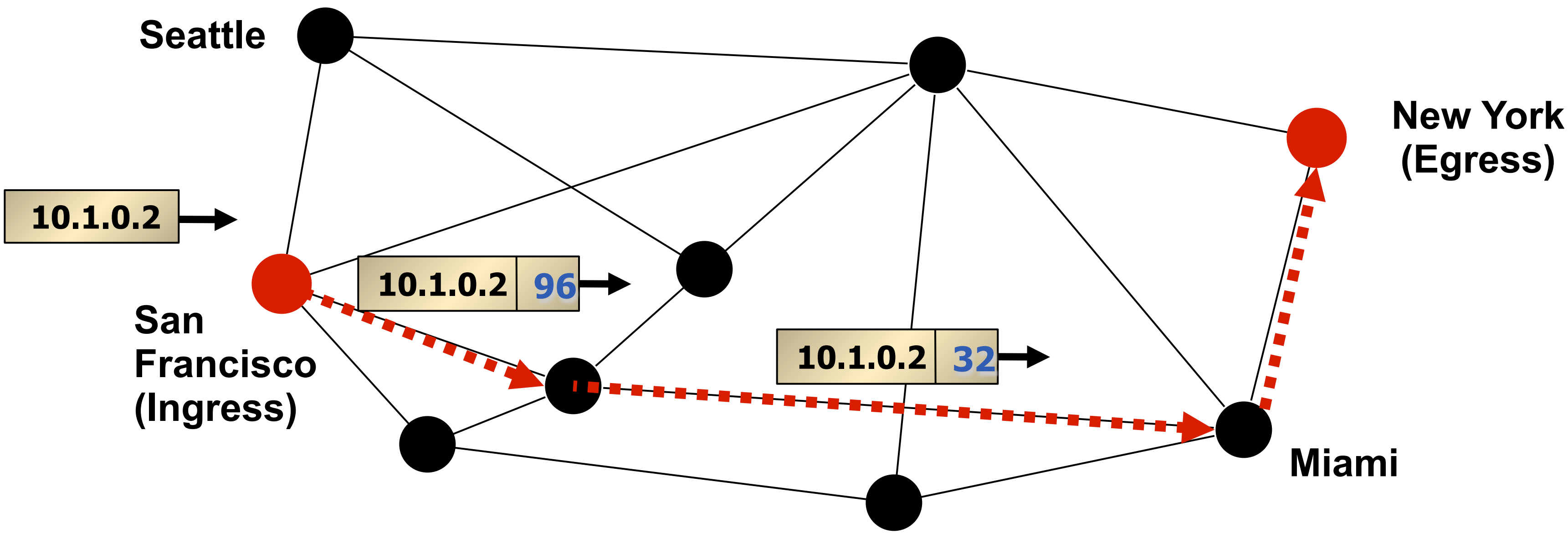
LDP Signaling Example: Forwarding

LDP path available to egress



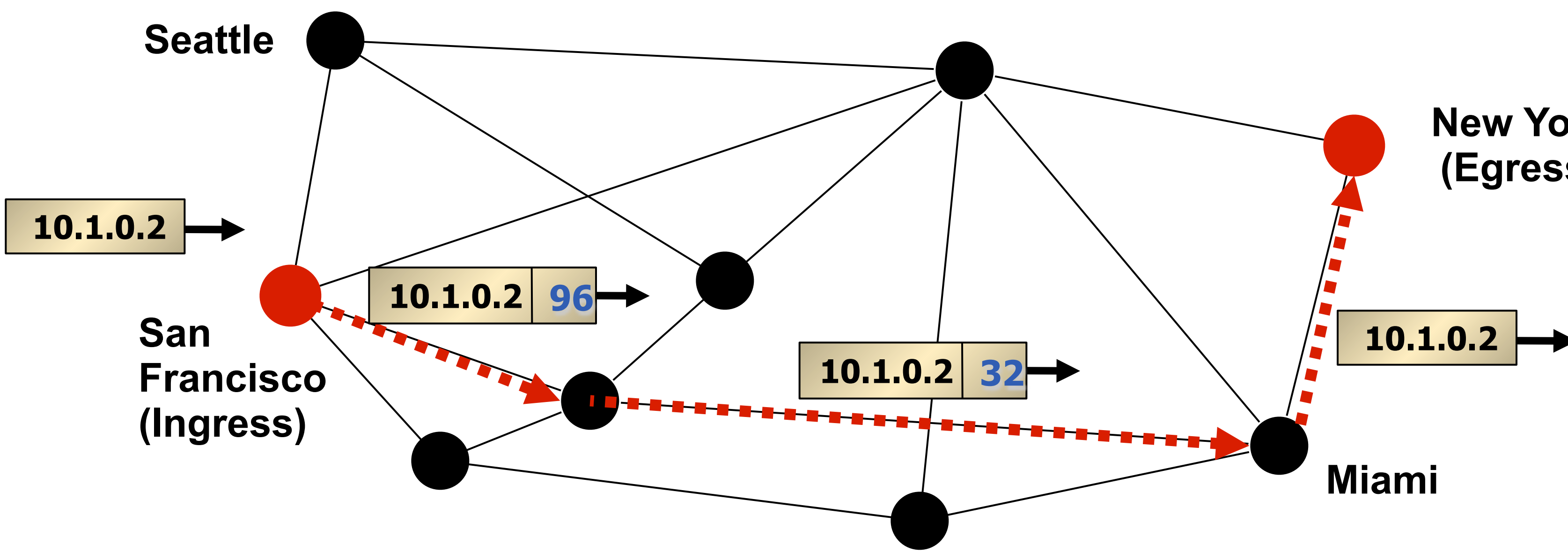
LDP Signaling Example: Forwarding

LDP path available to egress



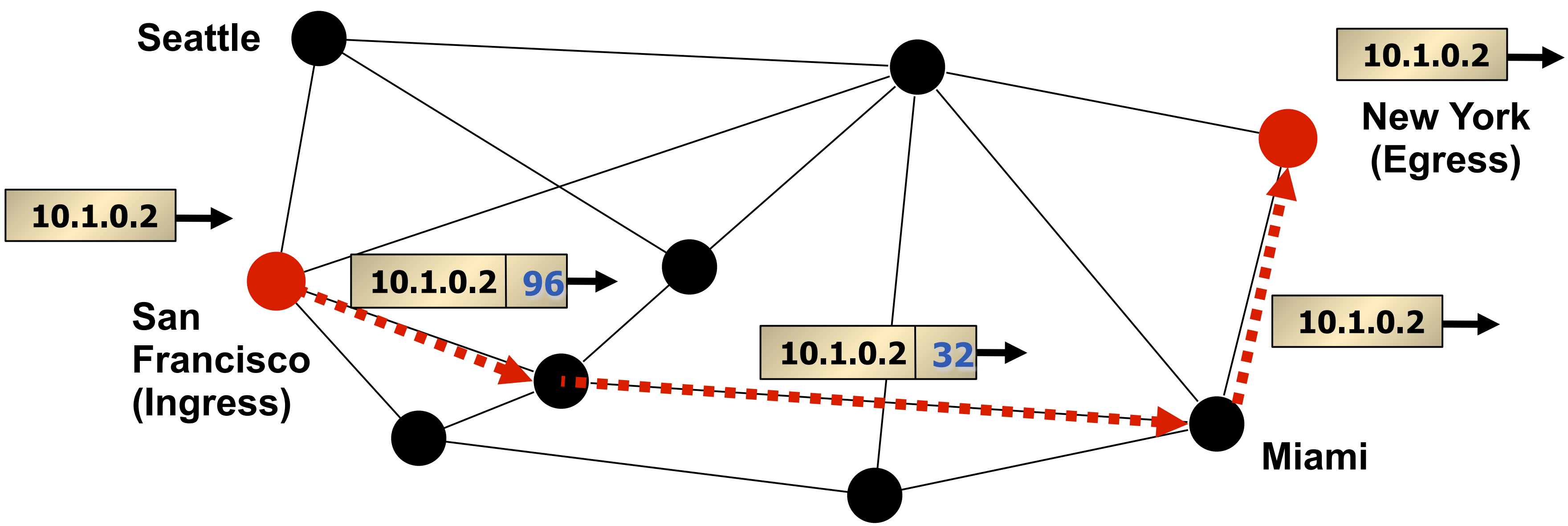
LDP Signaling Example: Forwarding

LDP path available to egress



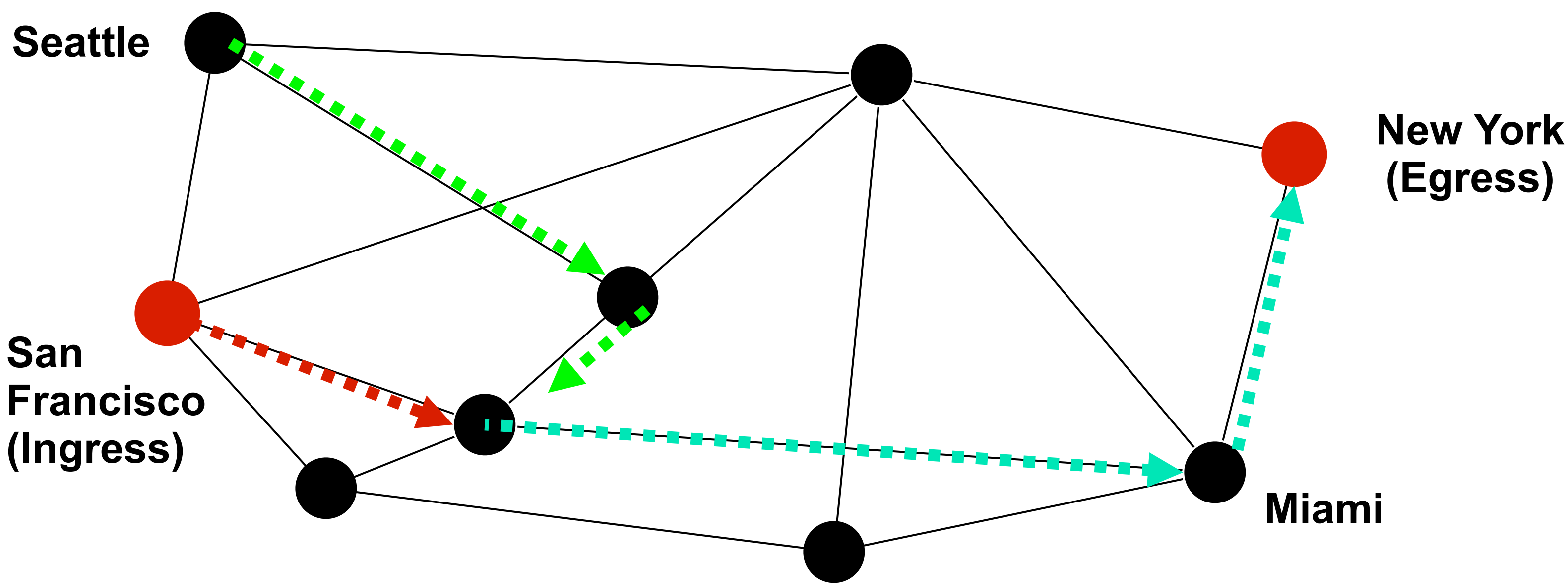
LDP Signaling Example: Forwarding

LDP path available to egress



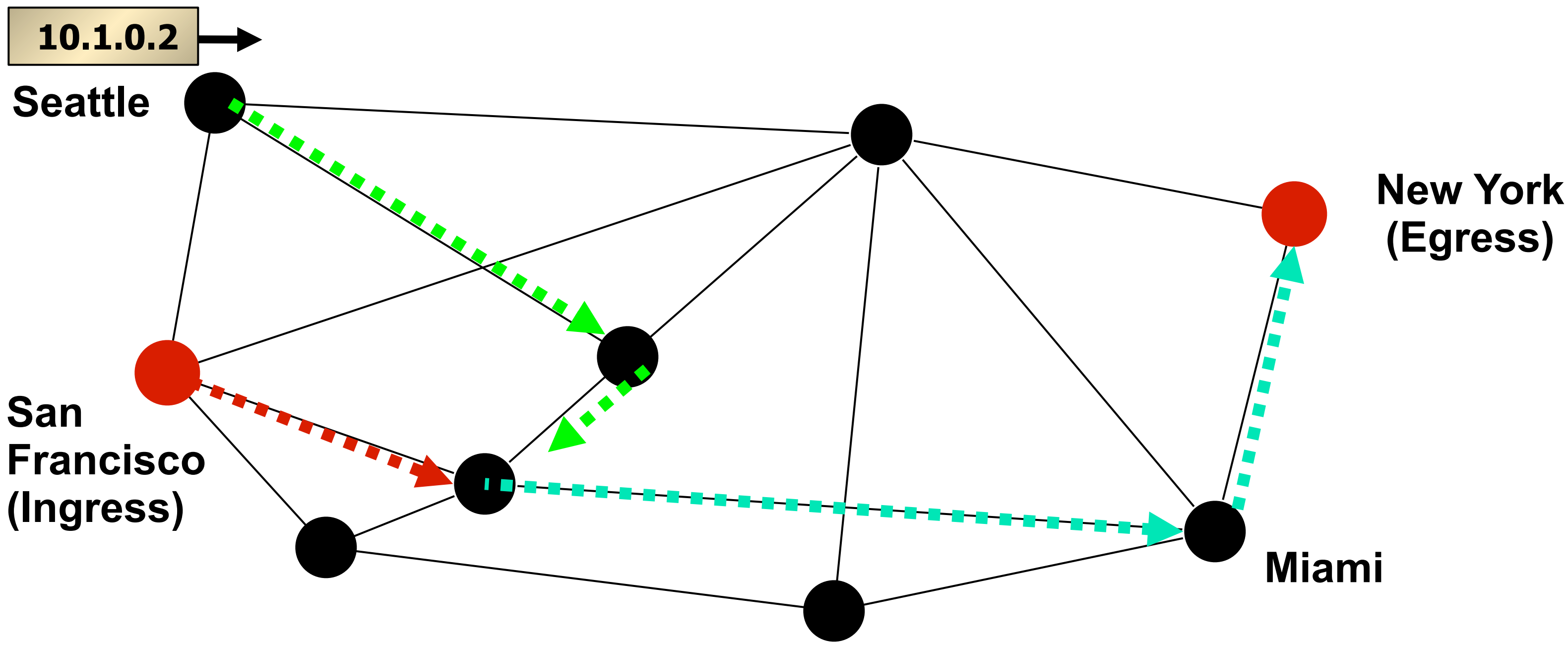
LDP Signaling Example:Forwarding 2

LSP Merging occurs



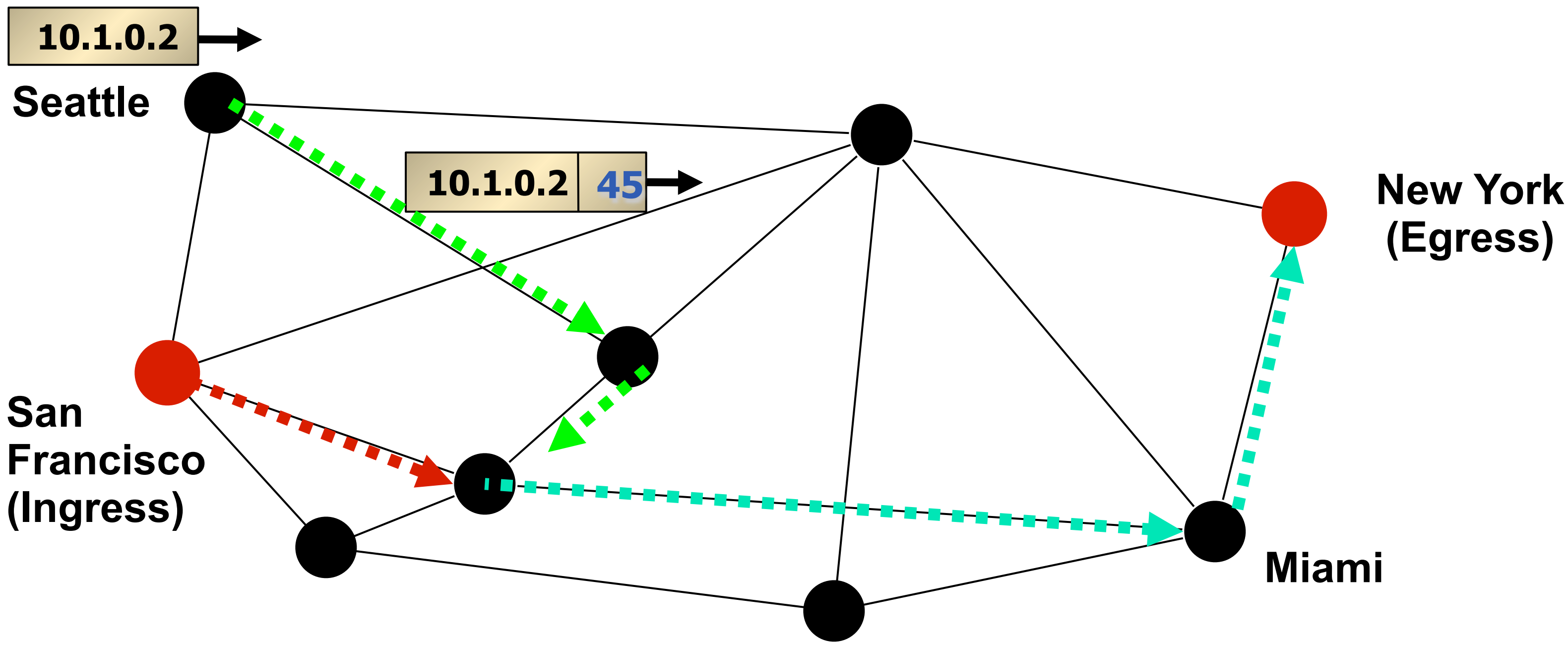
LDP Signaling Example:Forwarding 2

LSP Merging occurs



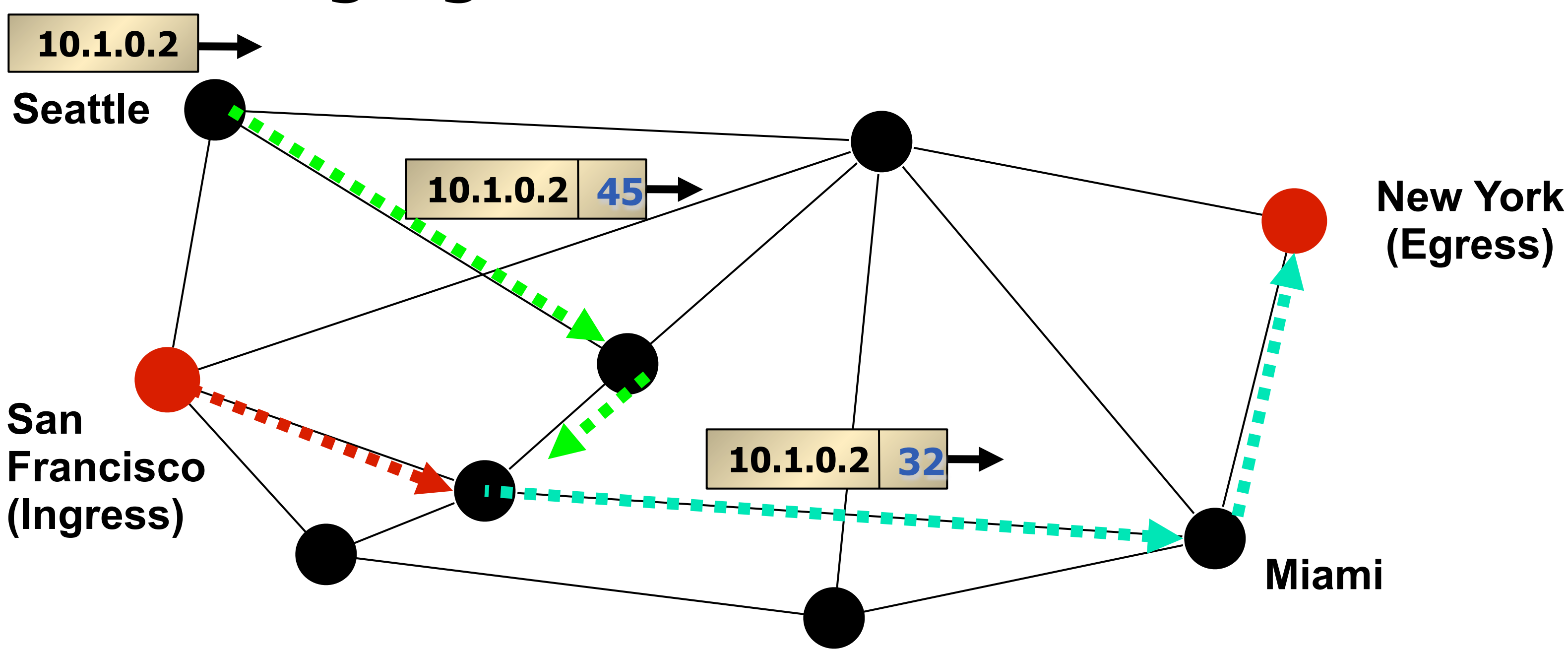
LDP Signaling Example:Forwarding 2

LSP Merging occurs



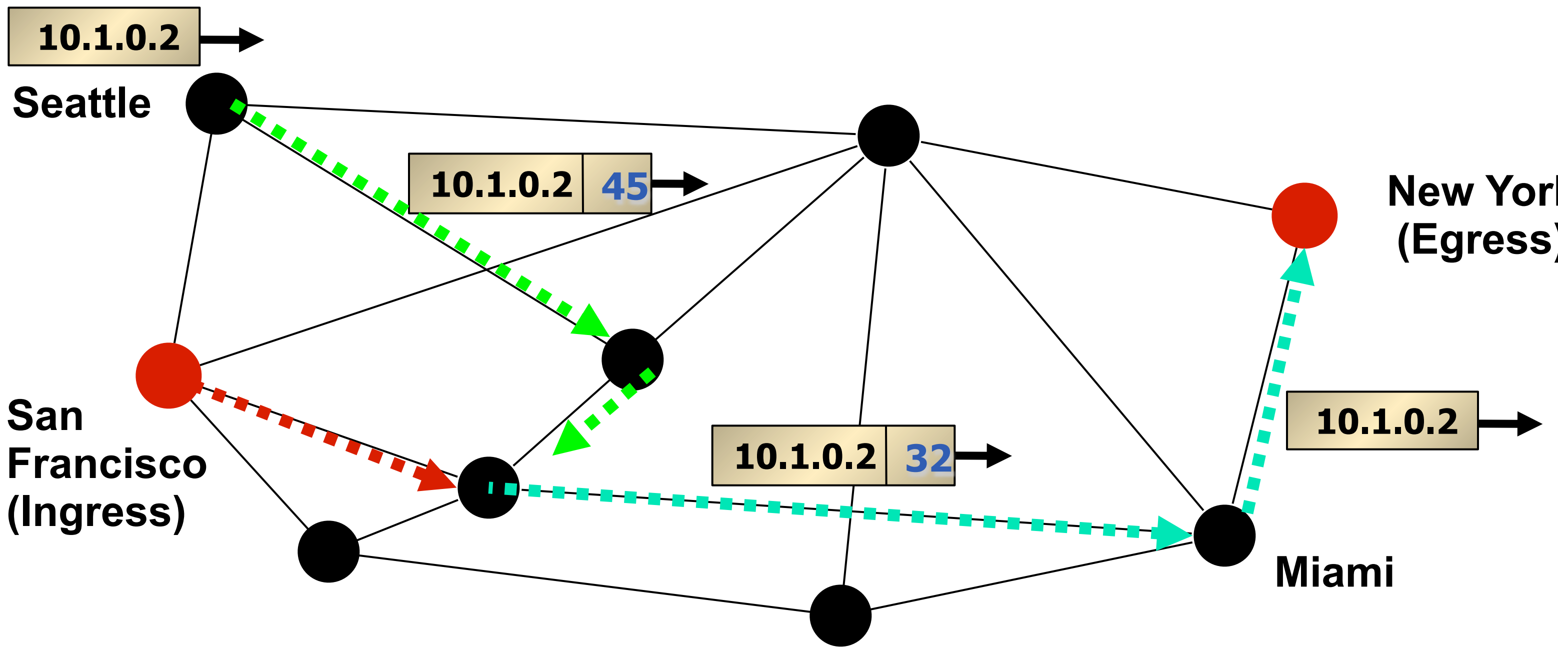
LDP Signaling Example:Forwarding 2

LSP Merging occurs



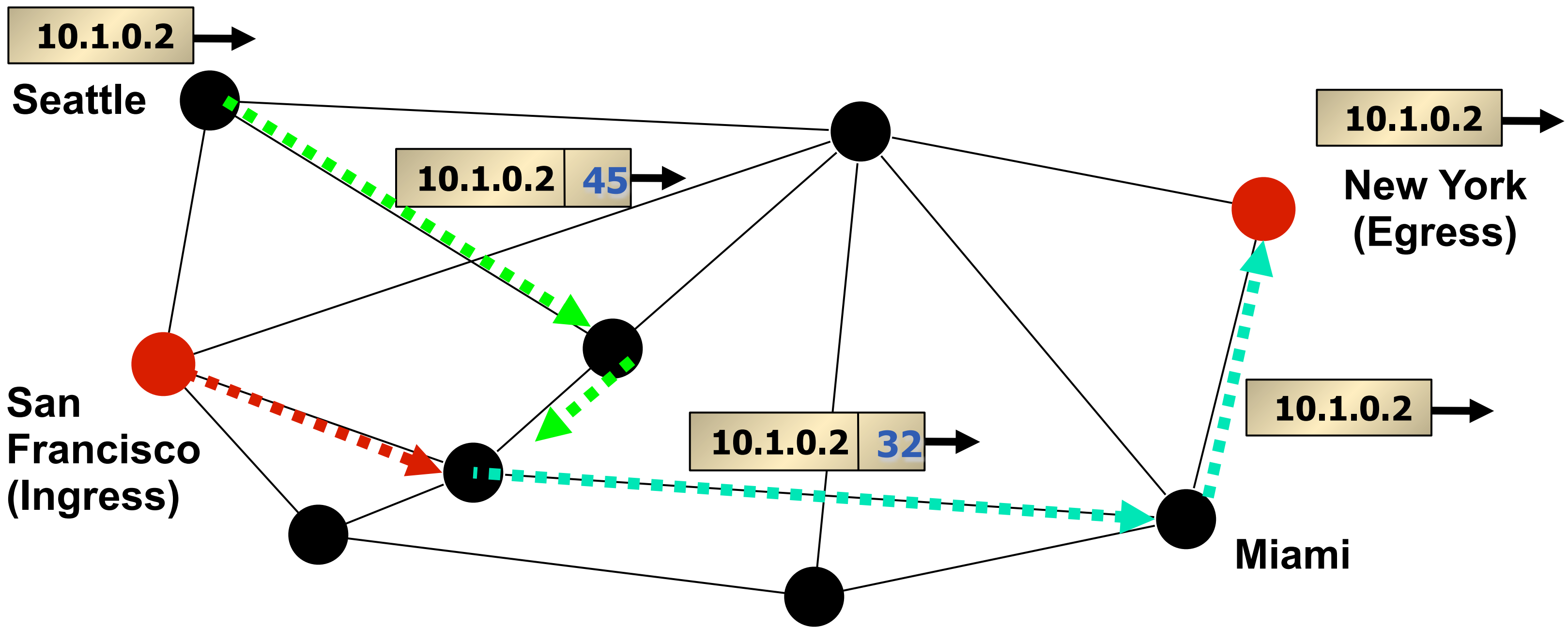
LDP Signaling Example:Forwarding 2

LSP Merging occurs



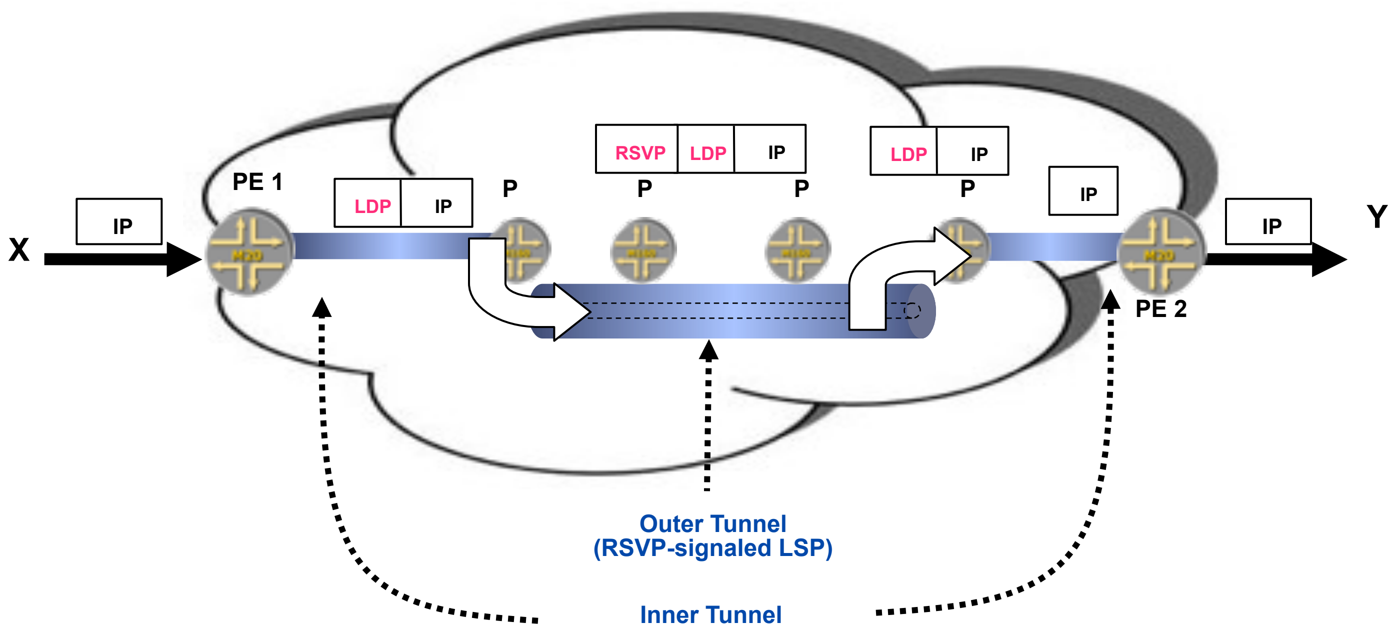
LDP Signaling Example:Forwarding 2

LSP Merging occurs



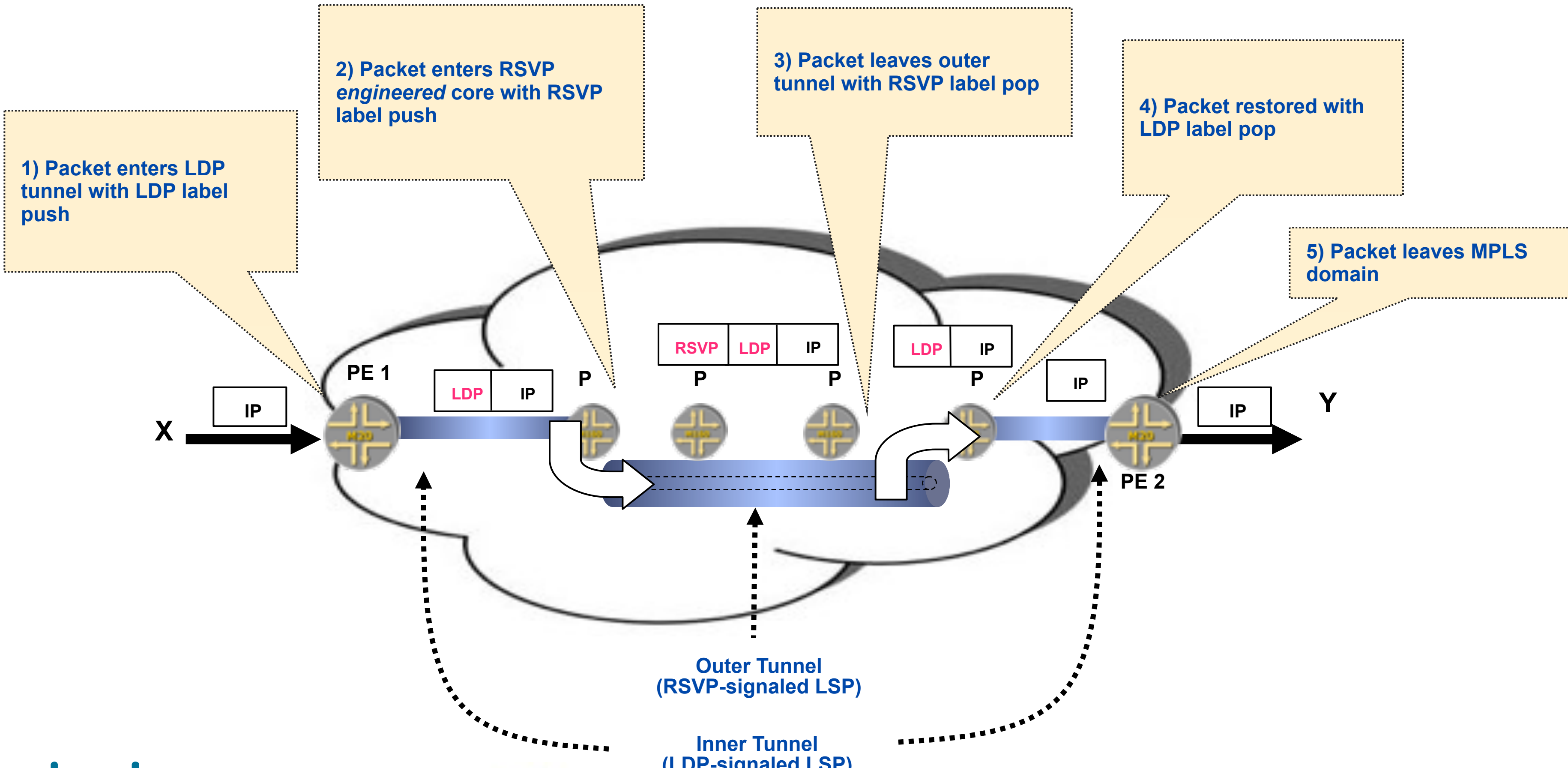
Label Stacking

- Label stacking improves scalability
  - Similar to ATM's VP and VC hierarchy



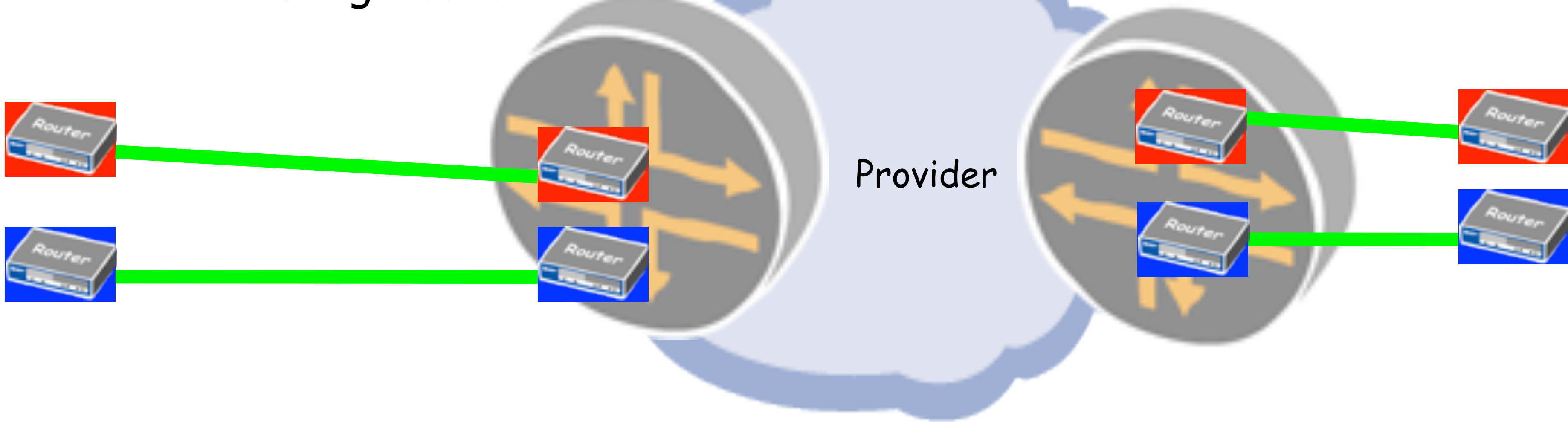
Label Stacking

- Label stacking improves scalability
  - Similar to ATM's VP and VC hierarchy



Layer 3 VPN's

- Now RFC 4364
  - RFC2547bis
- BGP/MPLS IP VPN's
- Other options
  - Virtual/logical Routers – simpler to understand perhaps, but scaling issues.



Influencing Deployment

- Cost ~2 x IP connectivity
- Expected to be 1:1 in 2-3 years

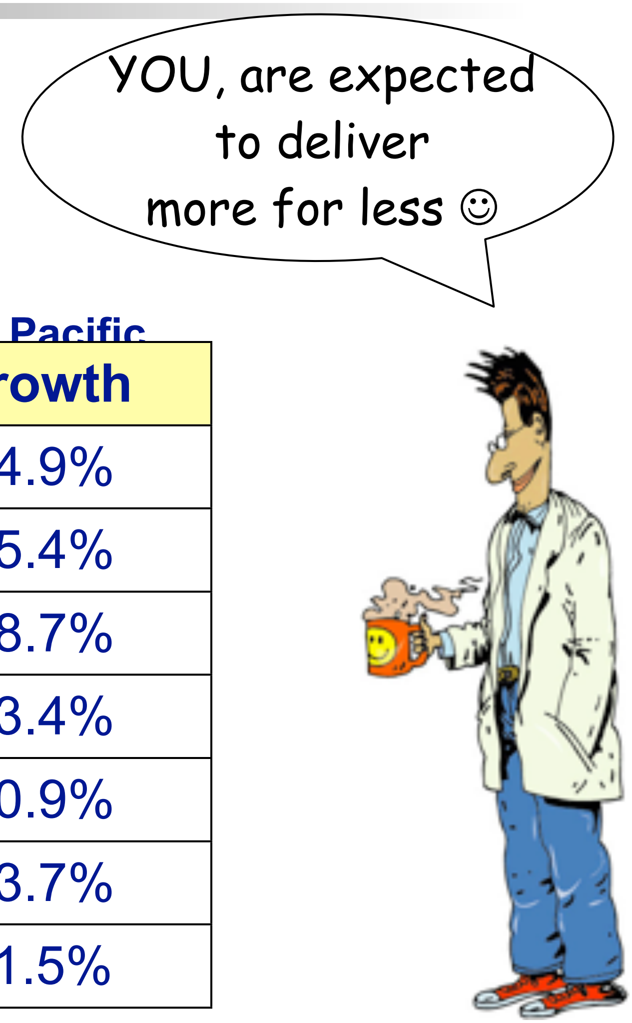
Predicted Revenue for IP VPN Services – Asia Pacific

Year	Revenues	Growth
2003	\$1.69b	24.9%
2004	\$2.11b	25.4%
2005	\$2.72b	28.7%
2006	\$3.36b	23.4%
2007	\$4.06b	20.9%
2008	\$4.62b	13.7%
2009	\$5.14b	11.5%

Source: Frost & Sullivan IP VPN Market: Revenue Forecasts (Asia Pacific), 2003-2009

Influencing Deployment

- Cost ~2 x IP connectivity
- Expected to be 1:1 in 2-3 years



Predicted Revenue for IP VPN Services – Asia Pacific

Year	Revenues	Growth
2003	$1.69b	24.9%
2004	$2.11b	25.4%
2005	$2.72b	28.7%
2006	$3.36b	23.4%
2007	$4.06b	20.9%
2008	$4.62b	13.7%
2009	$5.14b	11.5%

Source: Frost & Sullivan IP VPN Market: Revenue Forecasts (Asia Pacific), 2003-2009

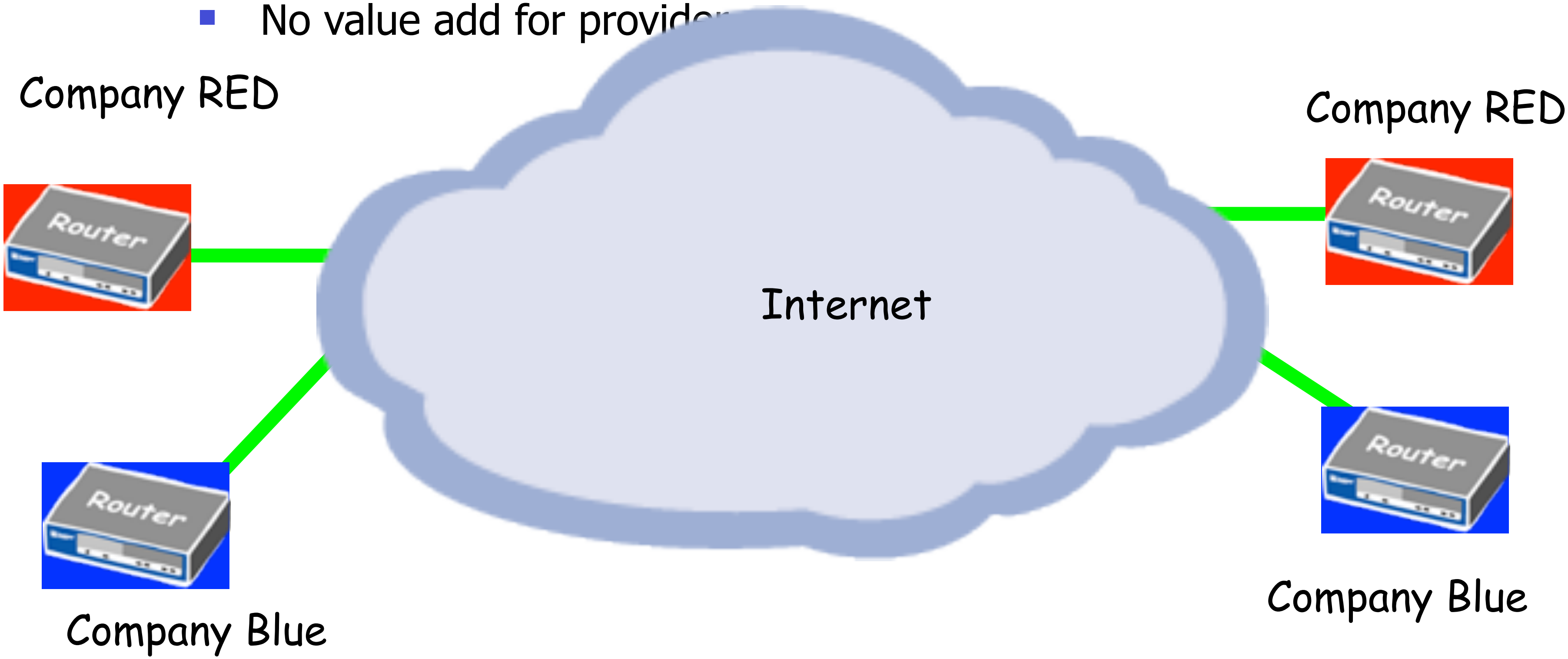
Layer 3 VPN’s (2547bis BGP/MPLS VPN’s)

Provider provisioned VPN

- ISP runs backbone for customer
  - Customer can be another ISP!
- Attractive to
  - Customer who do not want to run their own backbone
- Not attractive to
  - Customer who doesn’t trust carrier
  - Customers who’s jobs are threatened

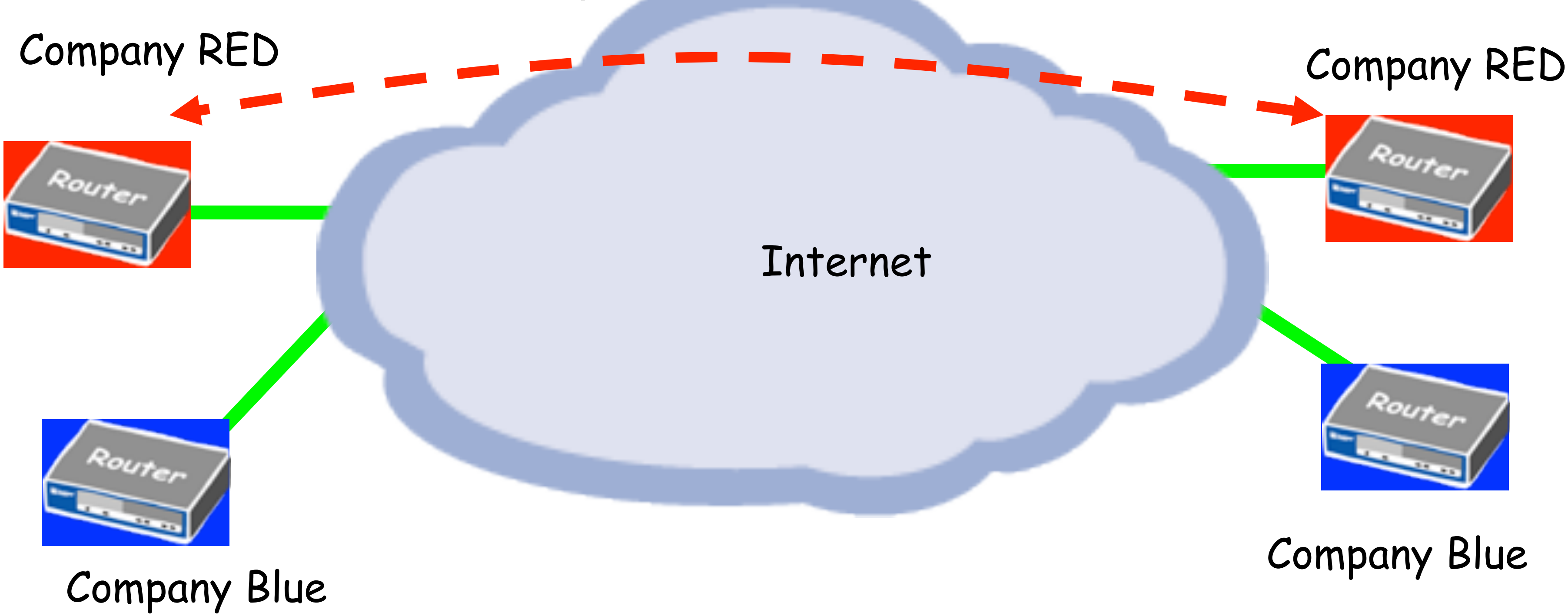
Traditional VPN's

- CPE based
- Customer controlled
- No value add for provider



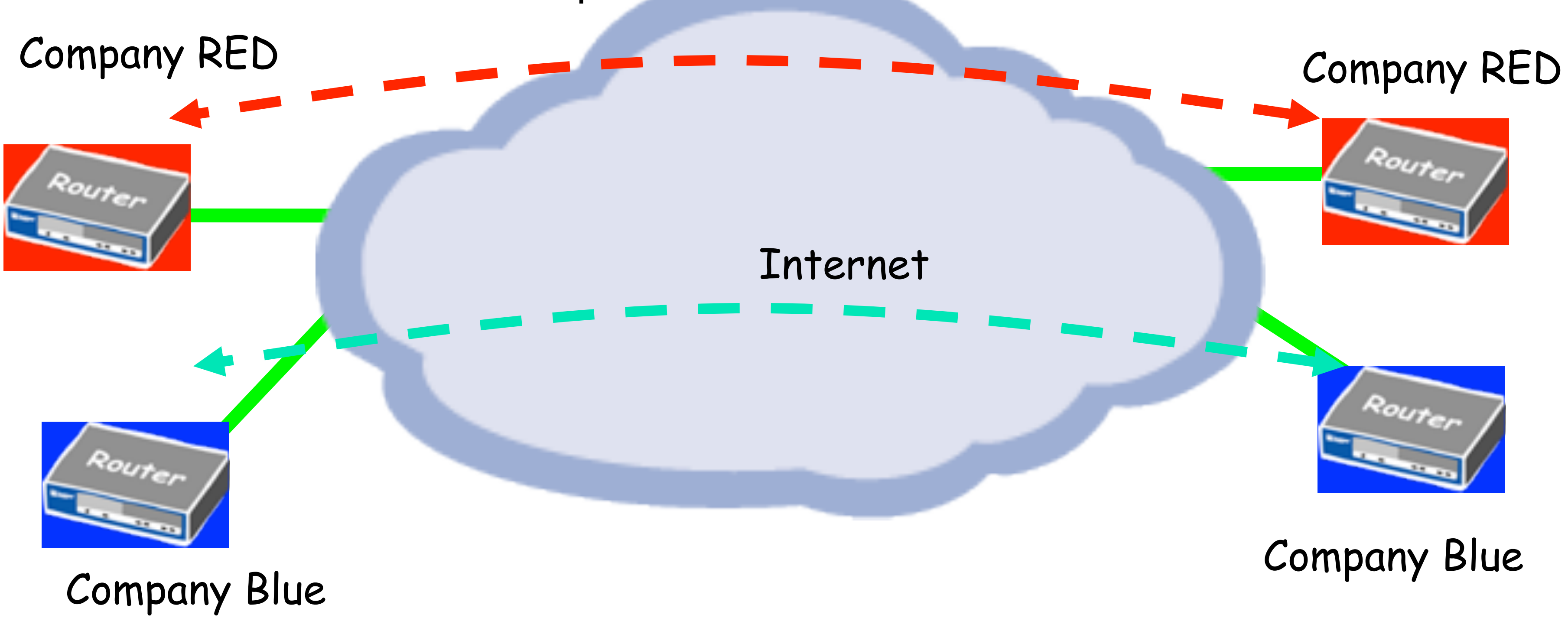
Traditional VPN’s

- CPE based
- Customer controlled
- No value add for provid



Traditional VPN's

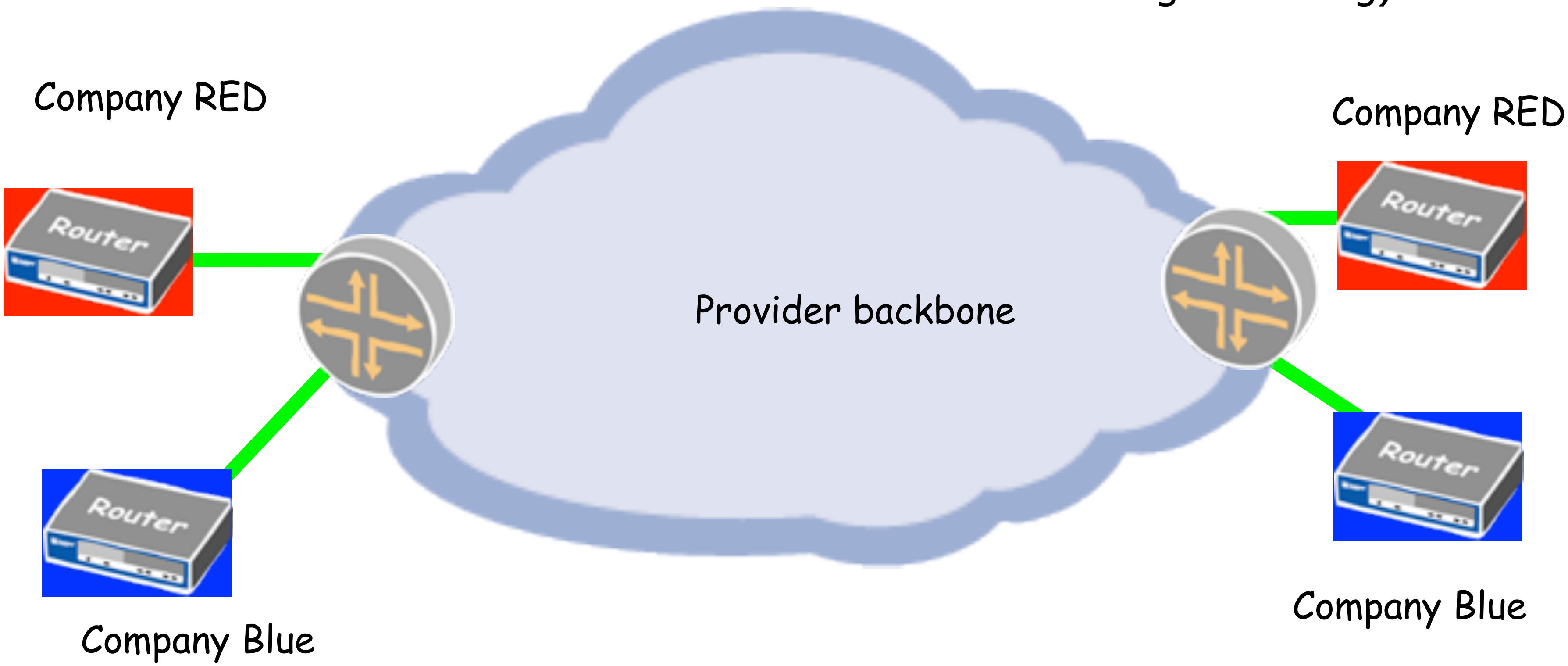
- CPE based
- Customer controlled
- No value add for provid



Provider provisioned VPN's - PPVPN

- PE based
- Customer outsource backbone

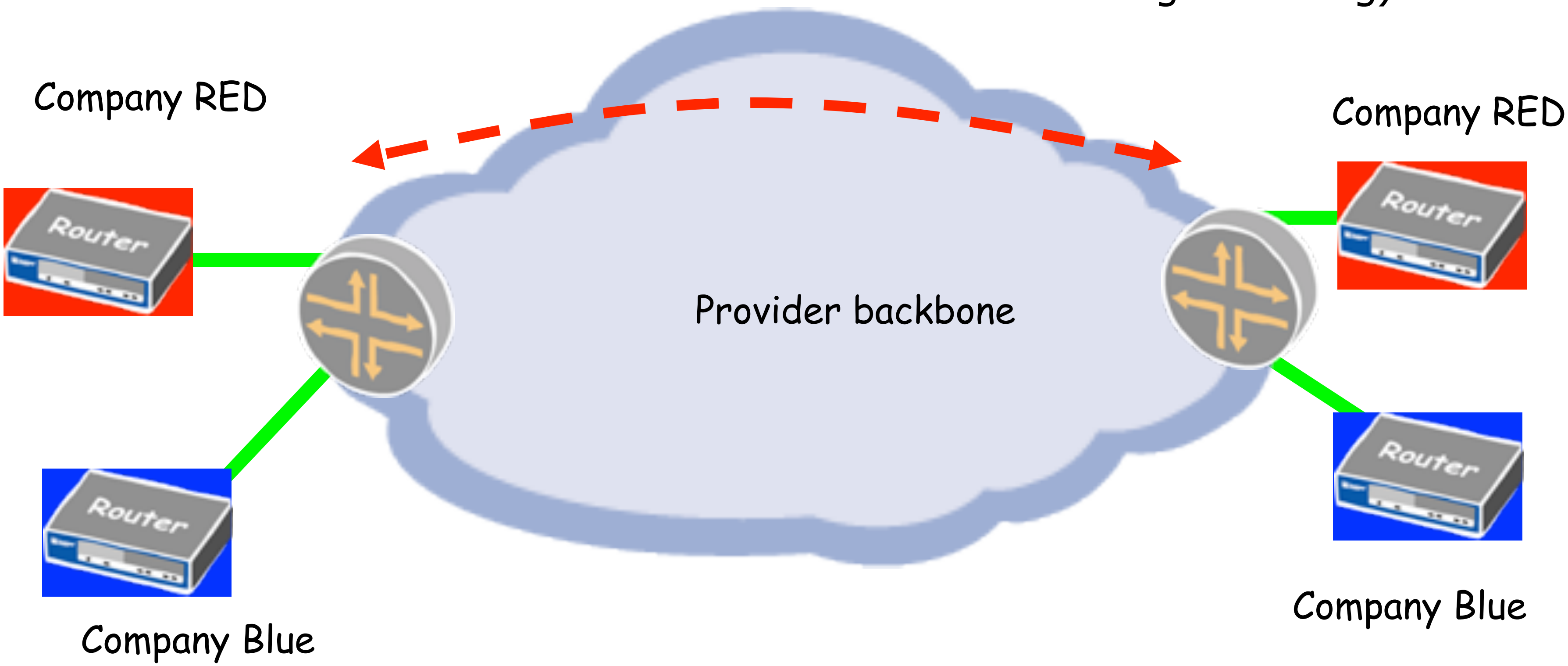
- Value add for provider
- Single Site Provisioning (BGP, + Route refresh + Route Target Filtering)



Provider provisioned VPN's - PPVPN

- PE based
- Customer outsource backbone

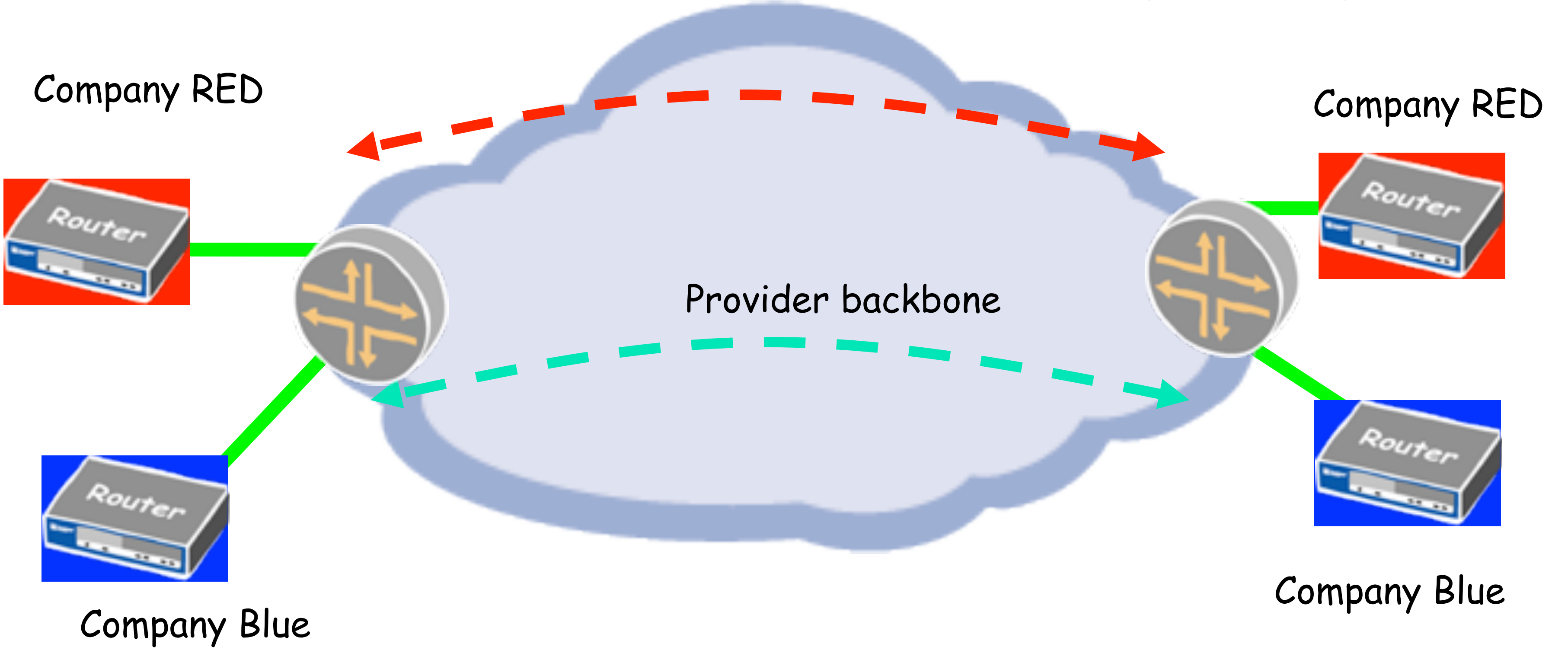
- Value add for provider
- Single Site Provisioning (BGP, + Route refresh + Route Target Filtering)



Provider provisioned VPN's - PPVPN

- PE based
- Customer outsource backbone

- Value add for provider
- Single Site Provisioning (BGP, + Route refresh + Route Target Filtering)



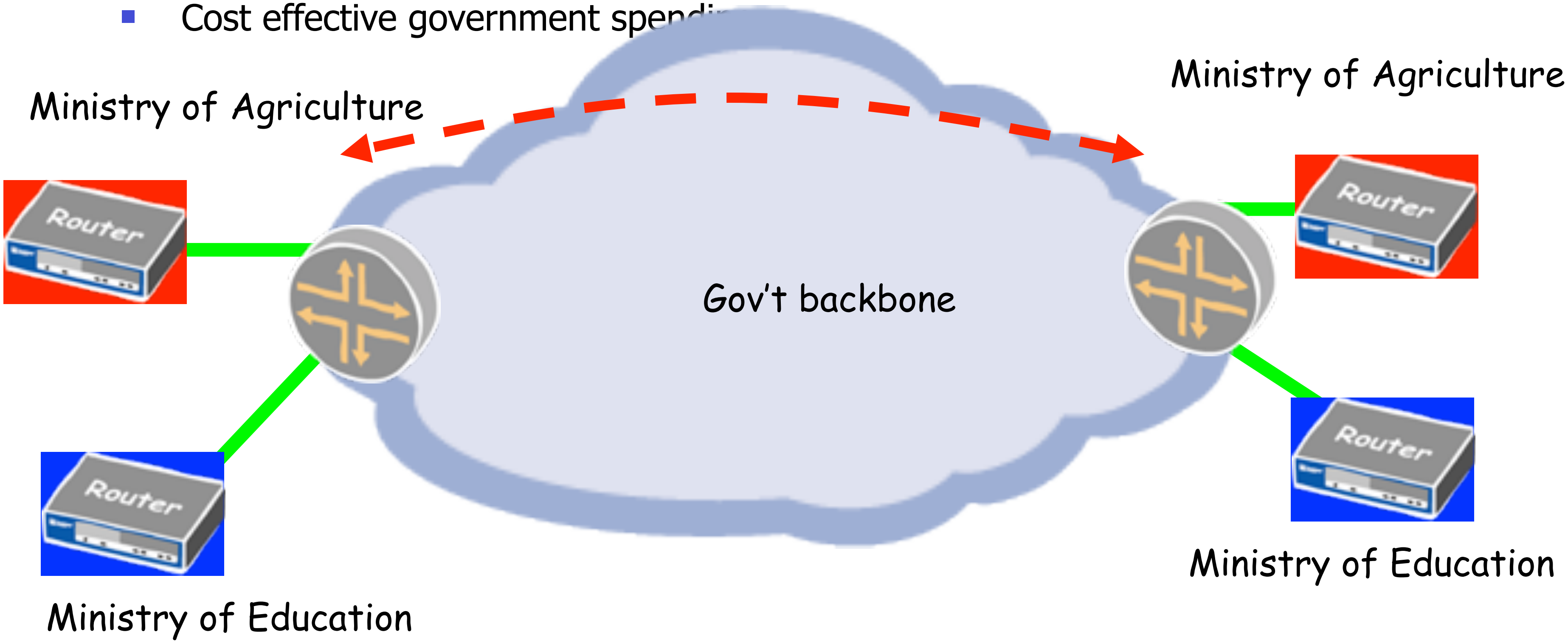
Sharing Network backbones

- Infrastructure built by one department
- Shared by other departments
- Cost effective government spendi

- Examples
  - Gov't backbones
  - Industry Aligned

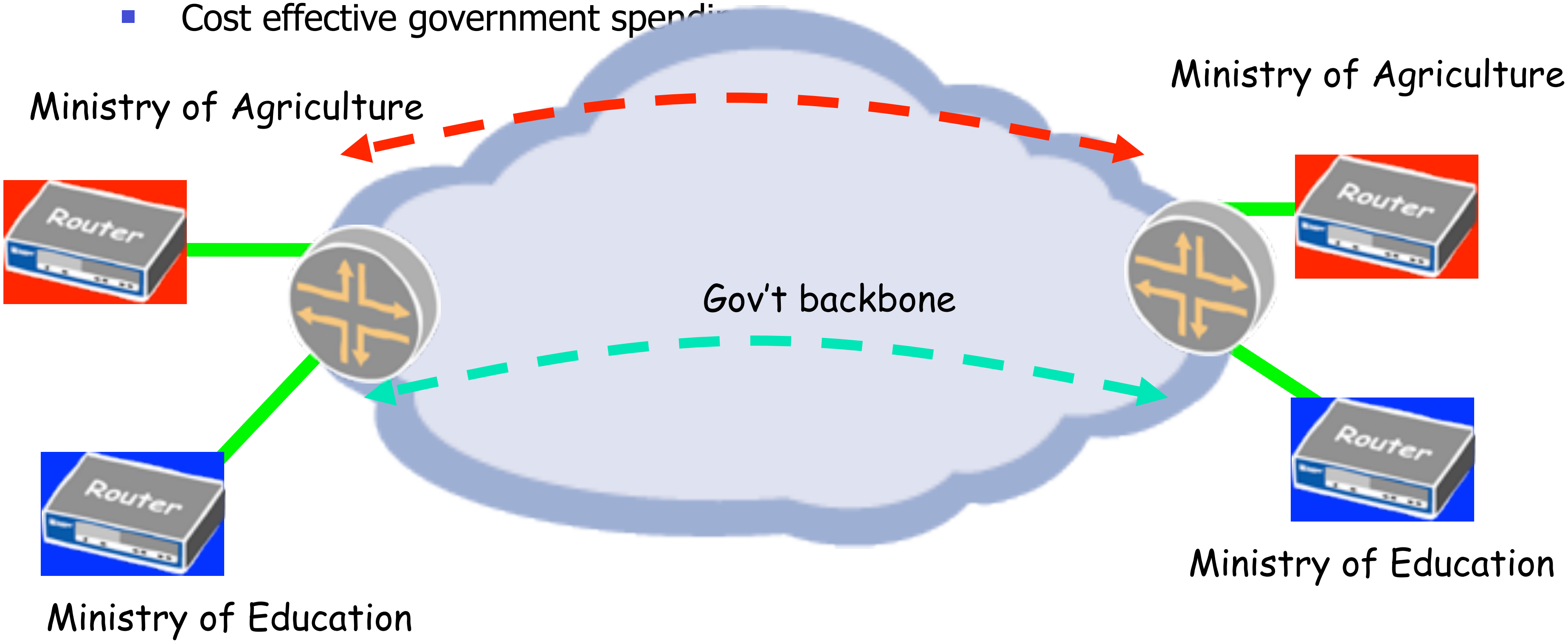
Sharing Network backbones

- Infrastructure built by one department
- Shared by other departments
- Cost effective government spendi...
- Examples
  - Gov't backbones
  - Industry Aligned



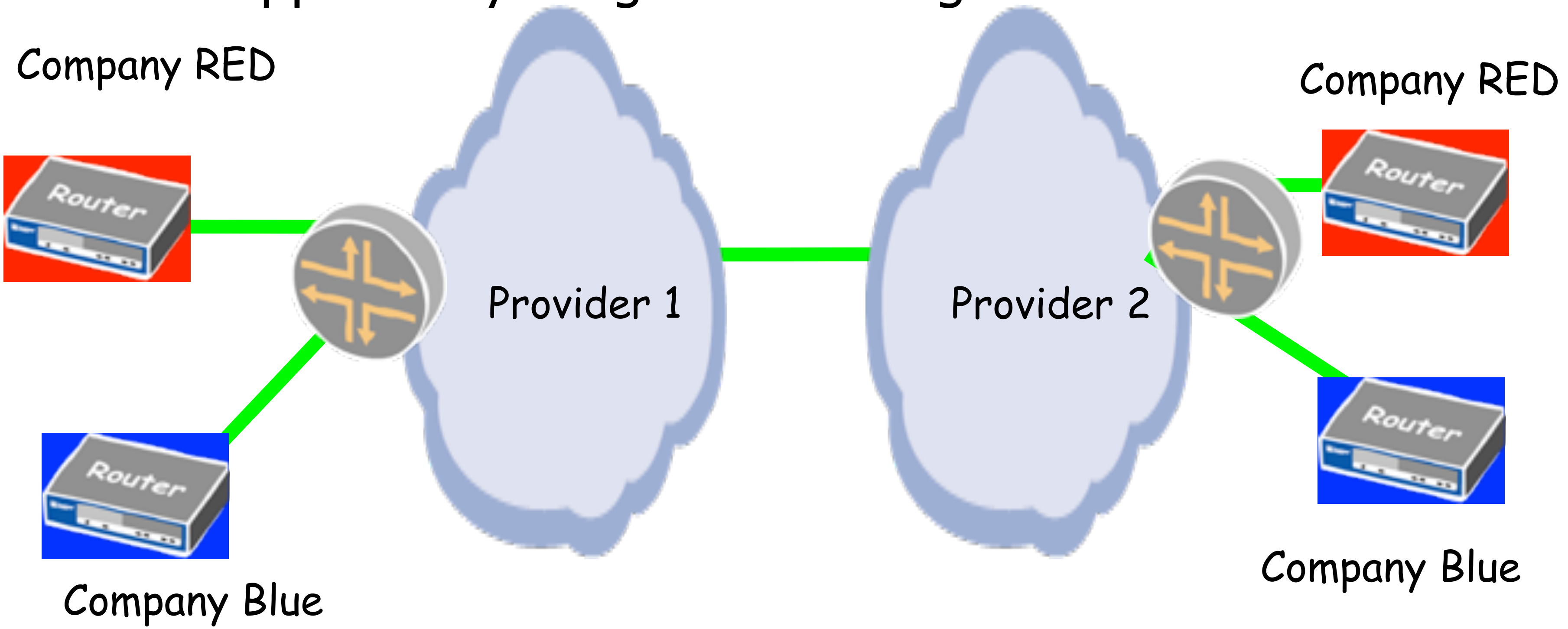
Sharing Network backbones

- Infrastructure built by one department
- Shared by other departments
- Cost effective government spendi[ng]
- Examples
  - Gov't backbones
  - Industry Aligned



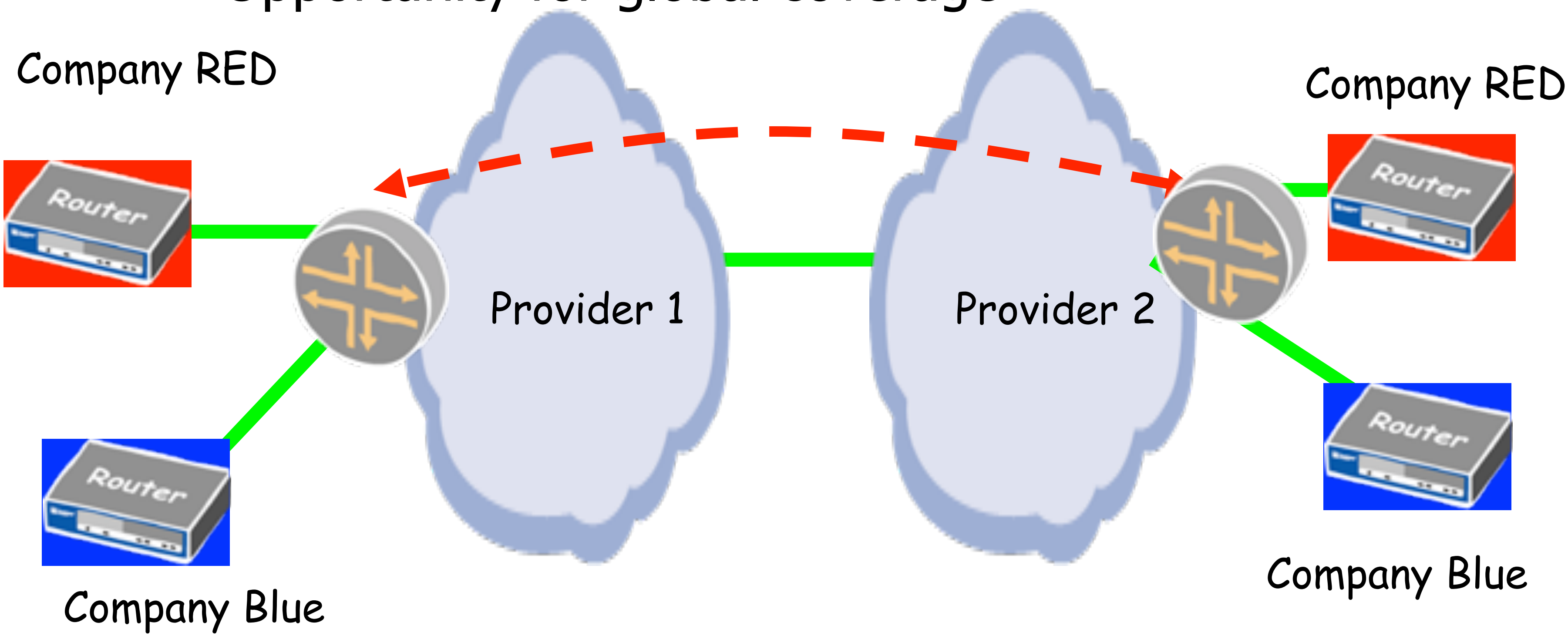
InterAS VPN's

- Requires Co-operation
- Opportunity for global coverage



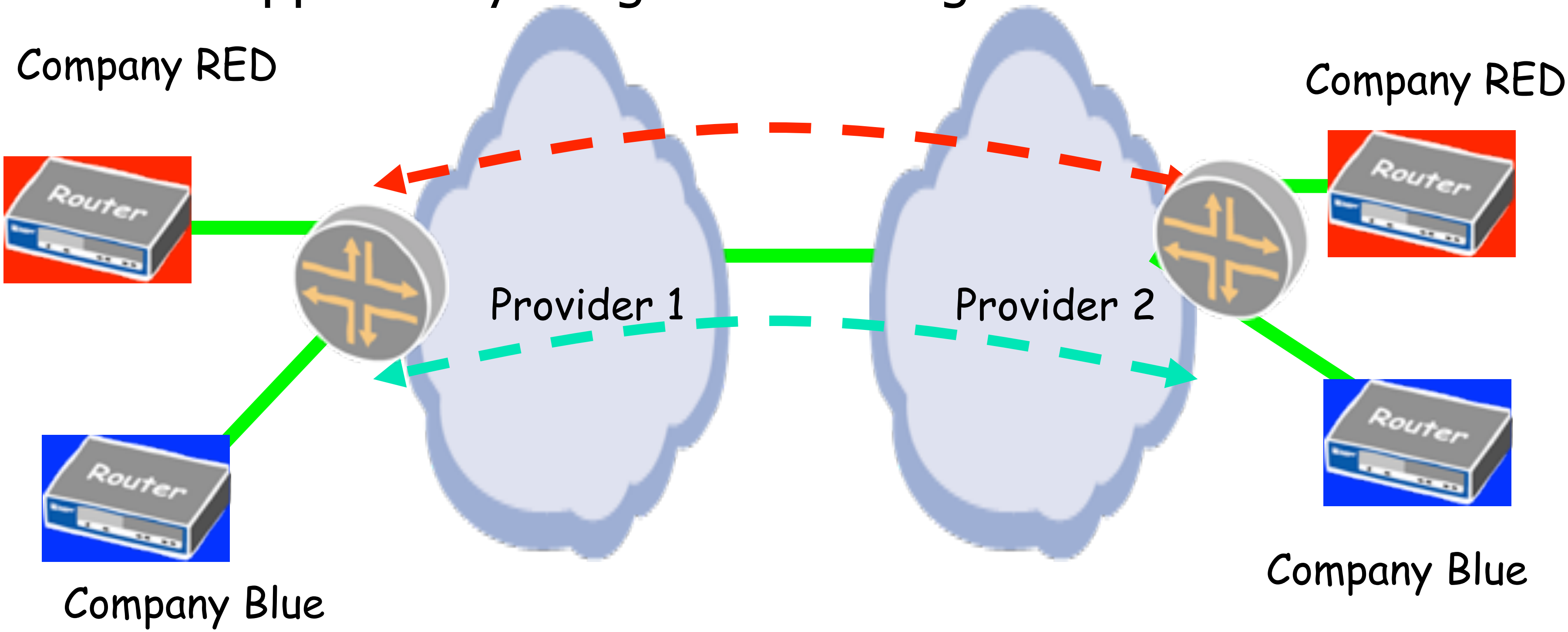
InterAS VPN's

- Requires Co-operation
- Opportunity for global coverage



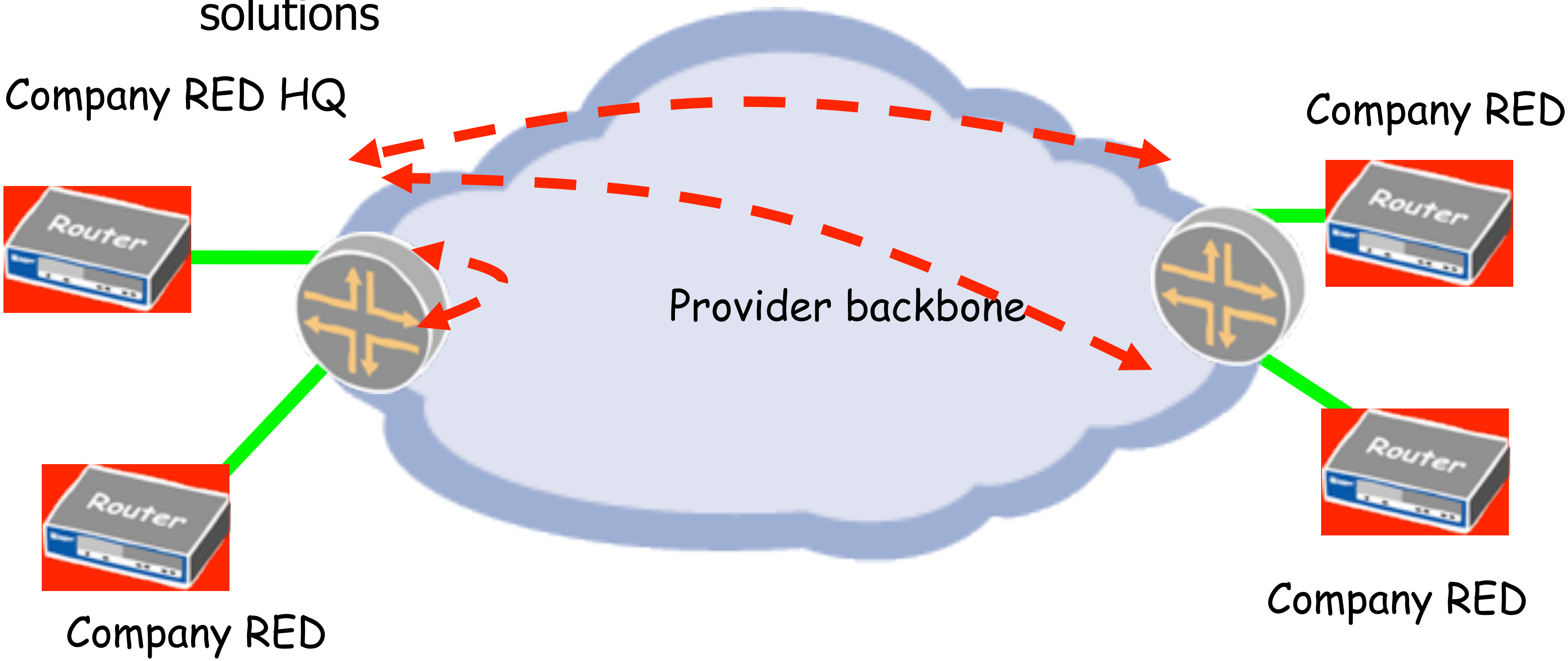
InterAS VPN's

- Requires Co-operation
- Opportunity for global coverage



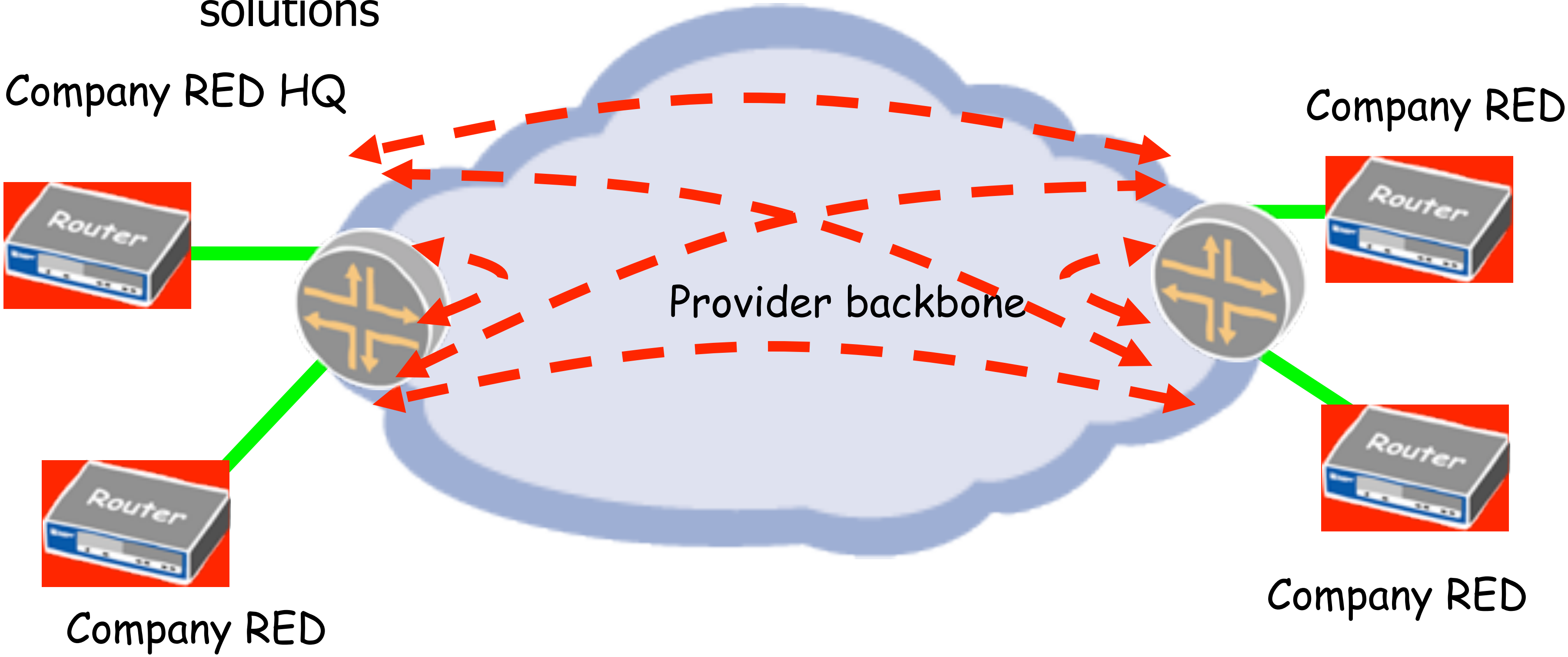
Site Connectivity

- Partial or Full Mesh is supported
- Full Mesh is more cost effective and competitive with traditional solutions



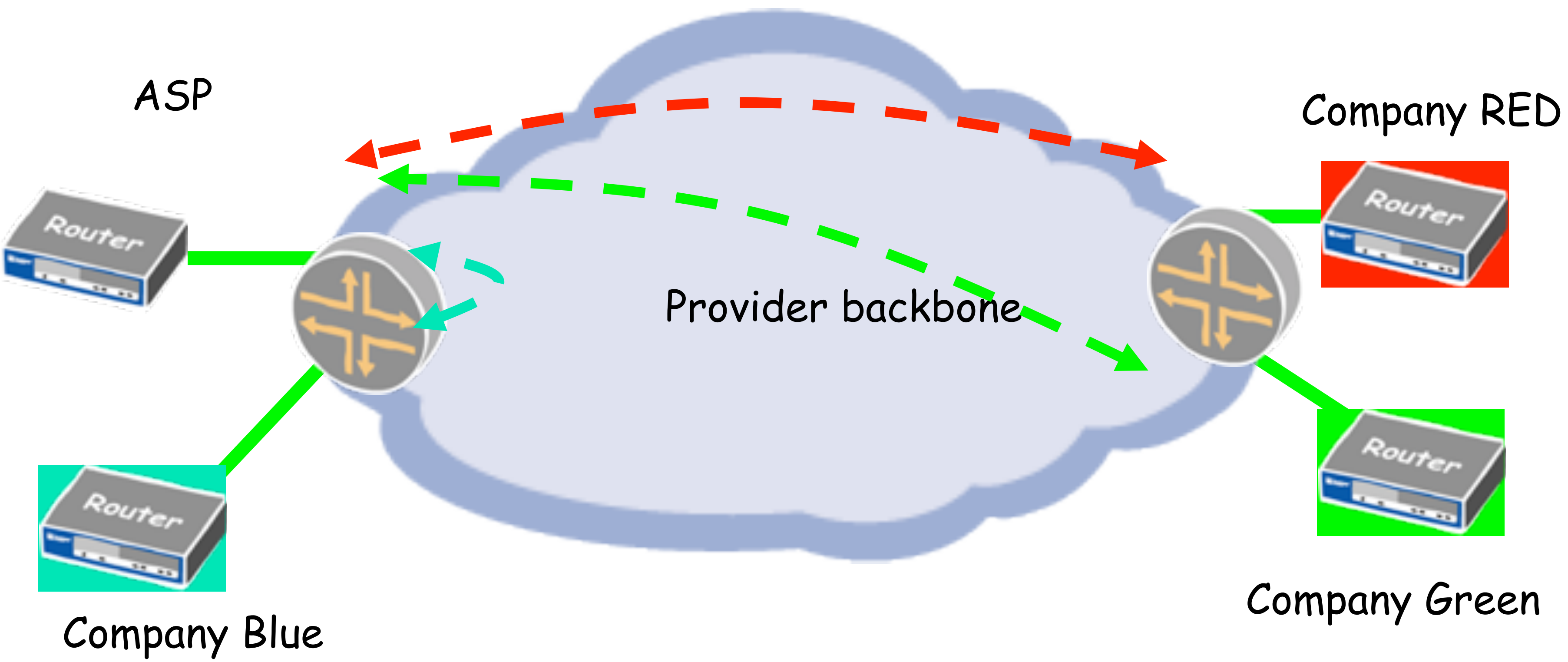
Site Connectivity

- Partial or Full Mesh is supported
- Full Mesh is more cost effective and competitive with traditional solutions



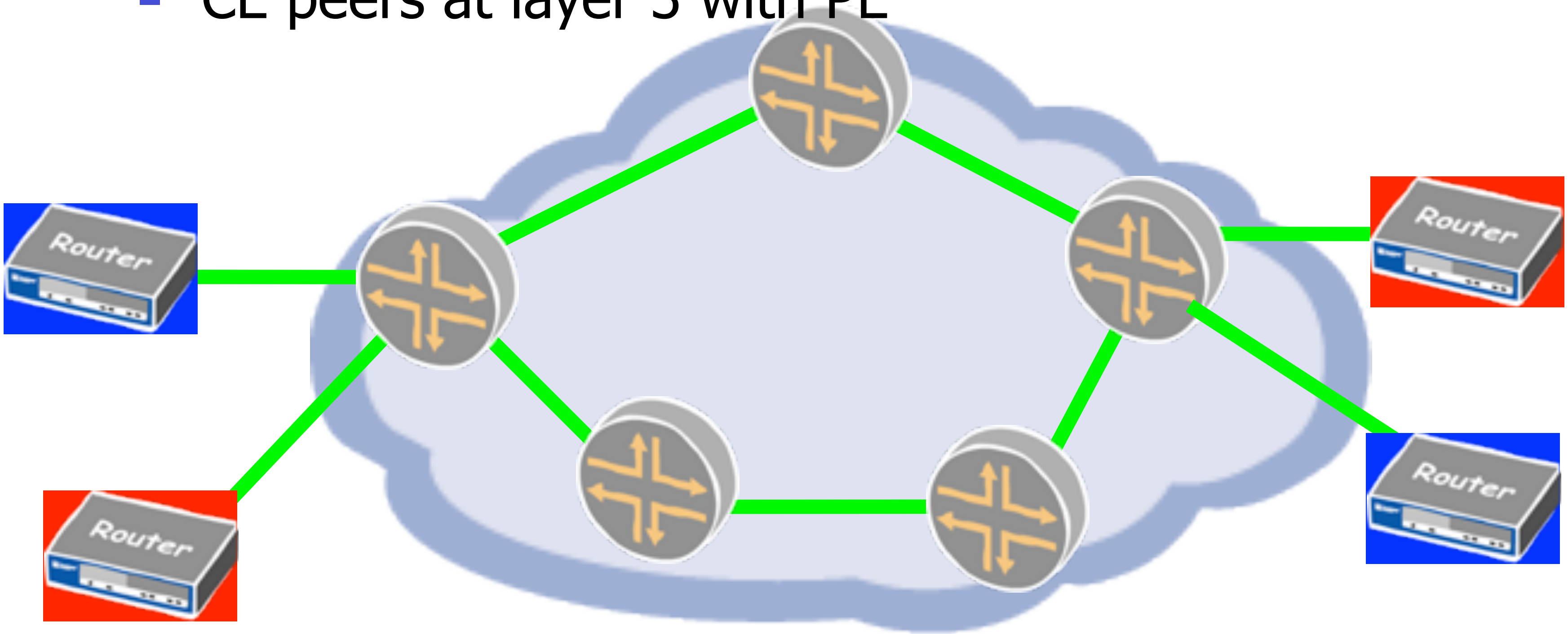
Overlapping VPN’s

- Suites application / service providers



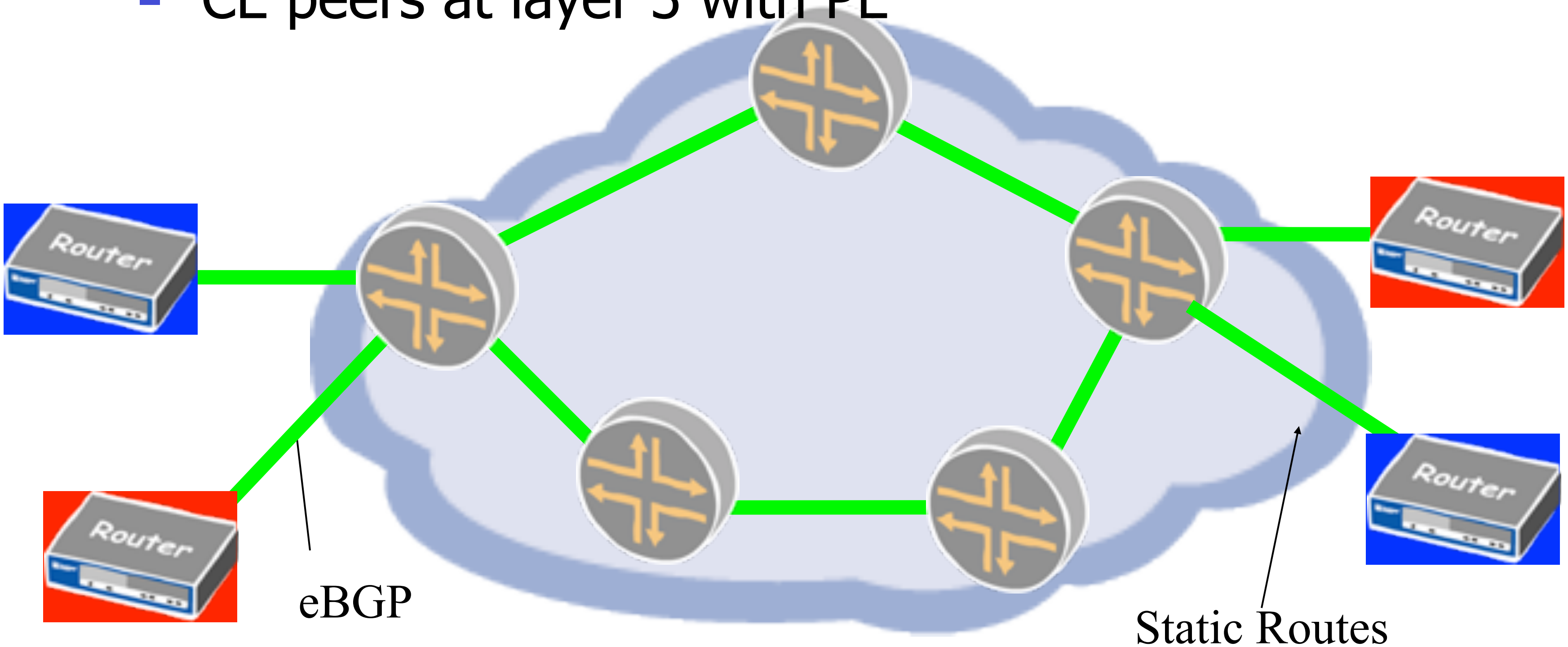
CE-PE interaction

- Any L2 connection, Any routing protocol
- CE peers at layer 3 with PE



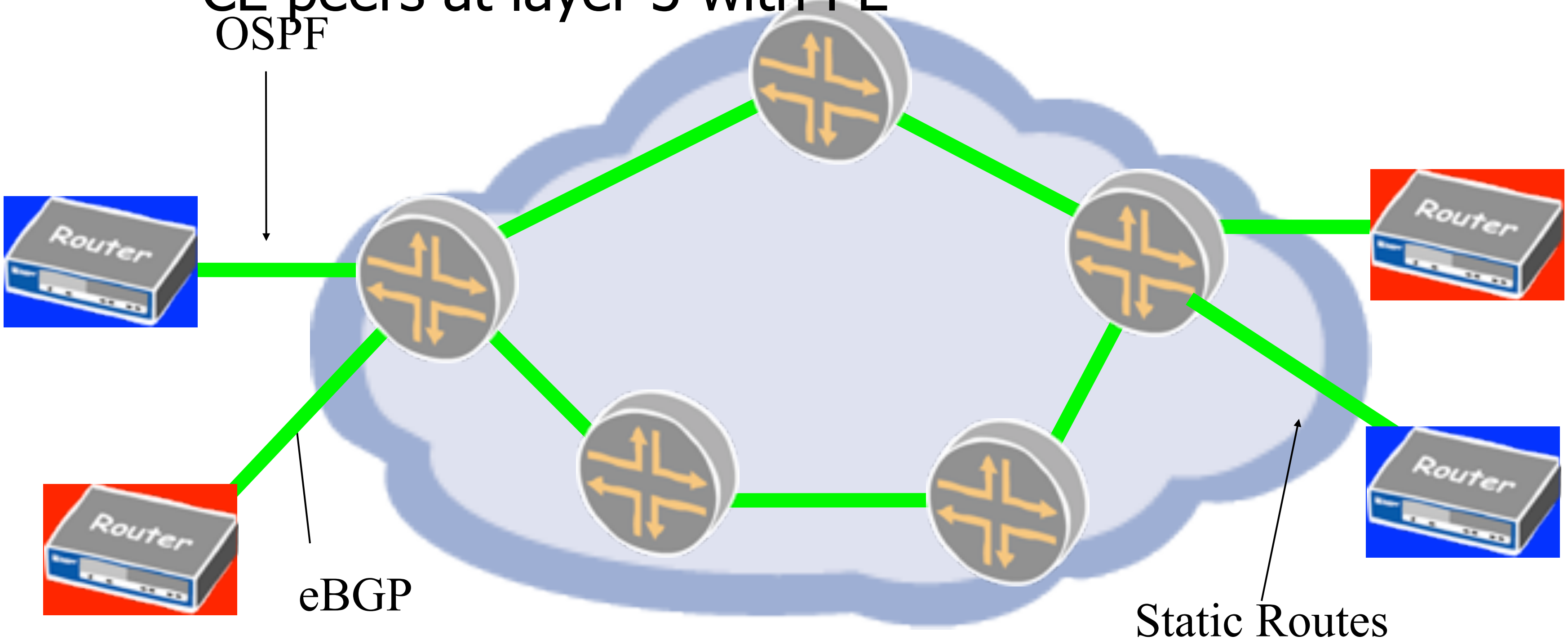
CE-PE interaction

- Any L2 connection, Any routing protocol
- CE peers at layer 3 with PE



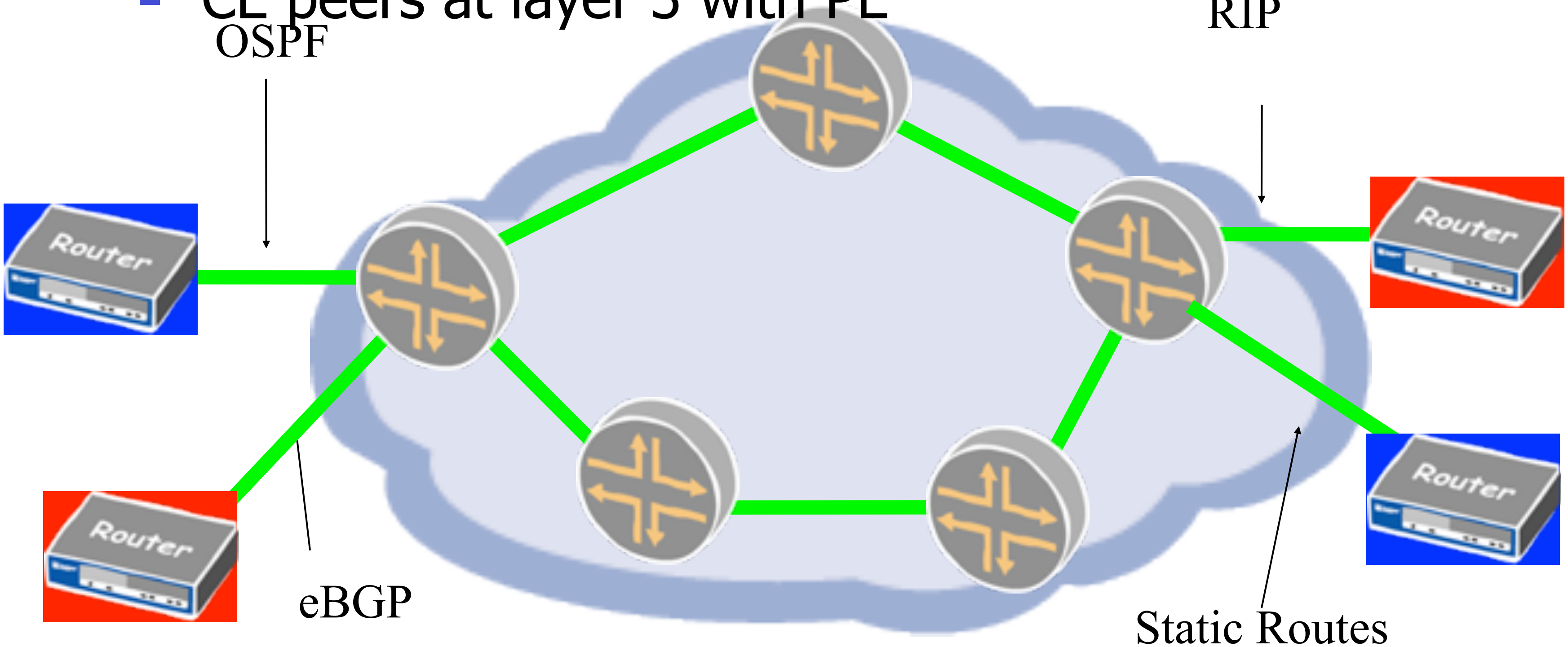
CE-PE interaction

- Any L2 connection, Any routing protocol
- CE peers at layer 3 with PE



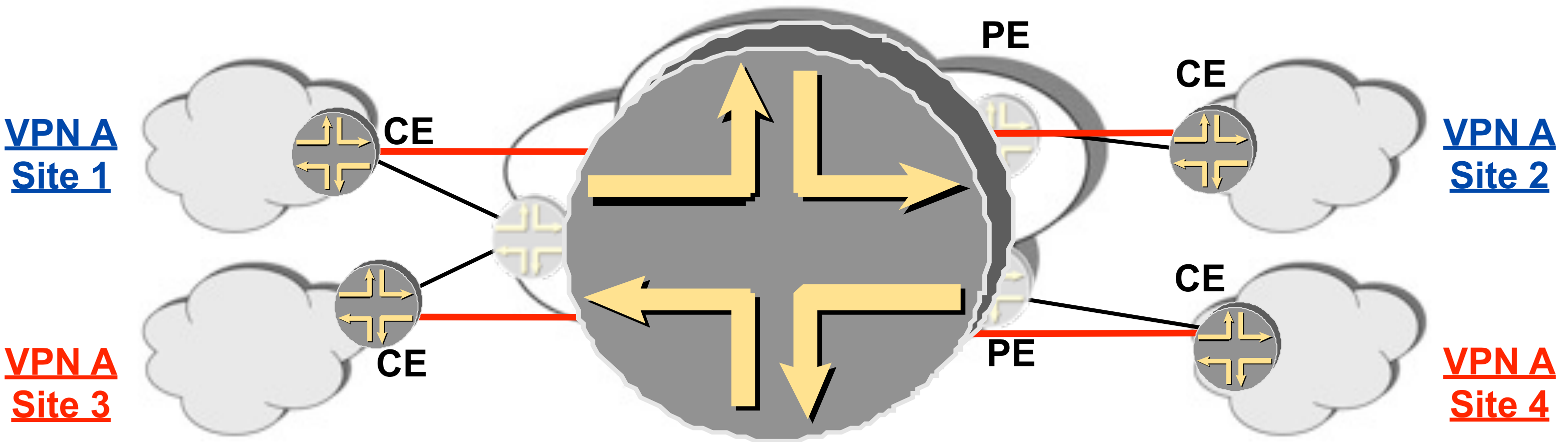
CE-PE interaction

- Any L2 connection, Any routing protocol
- CE peers at layer 3 with PE



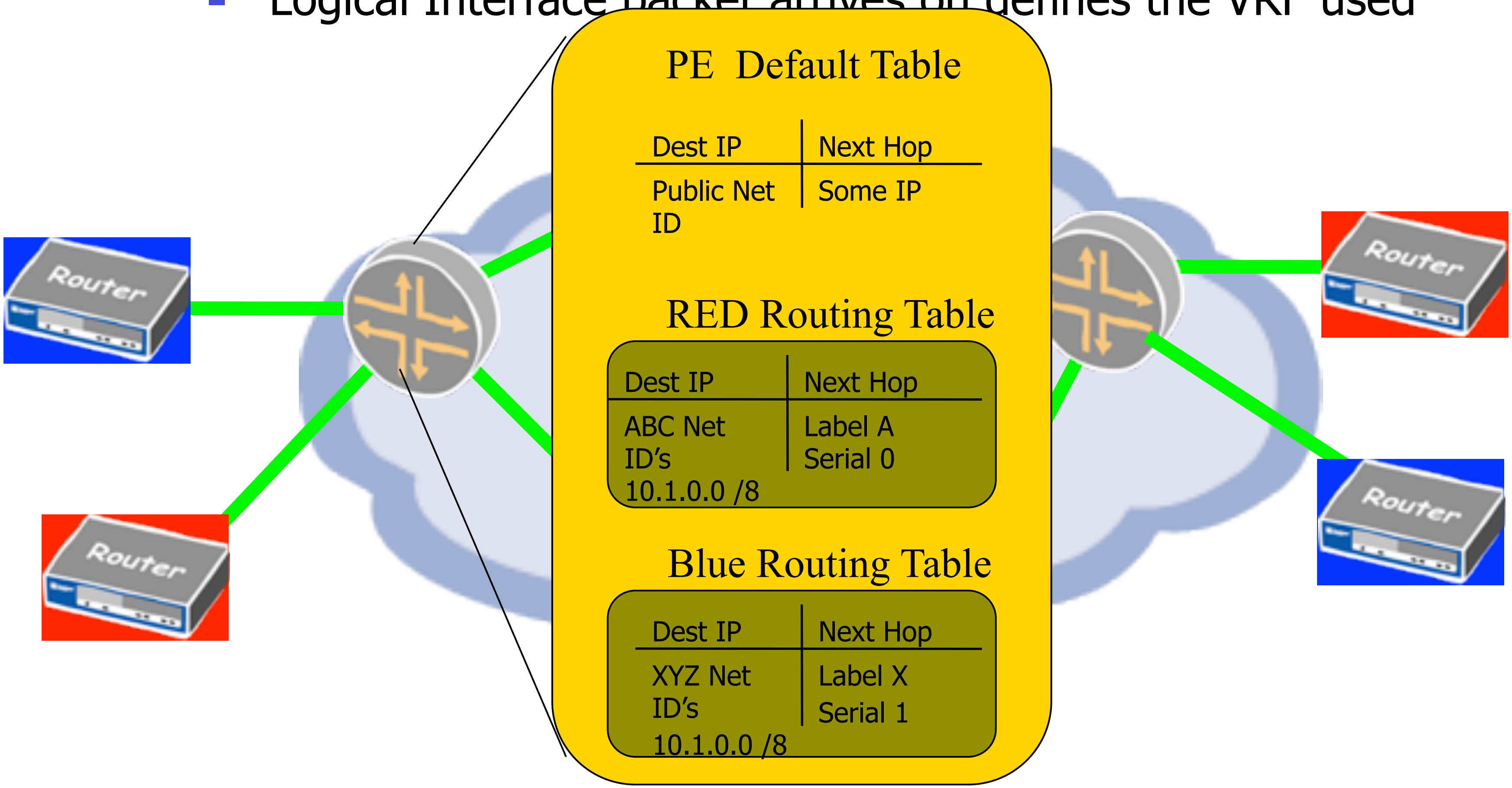
Customer View of L3VPN

- Make the cloud look like a router
- Single site provisioning



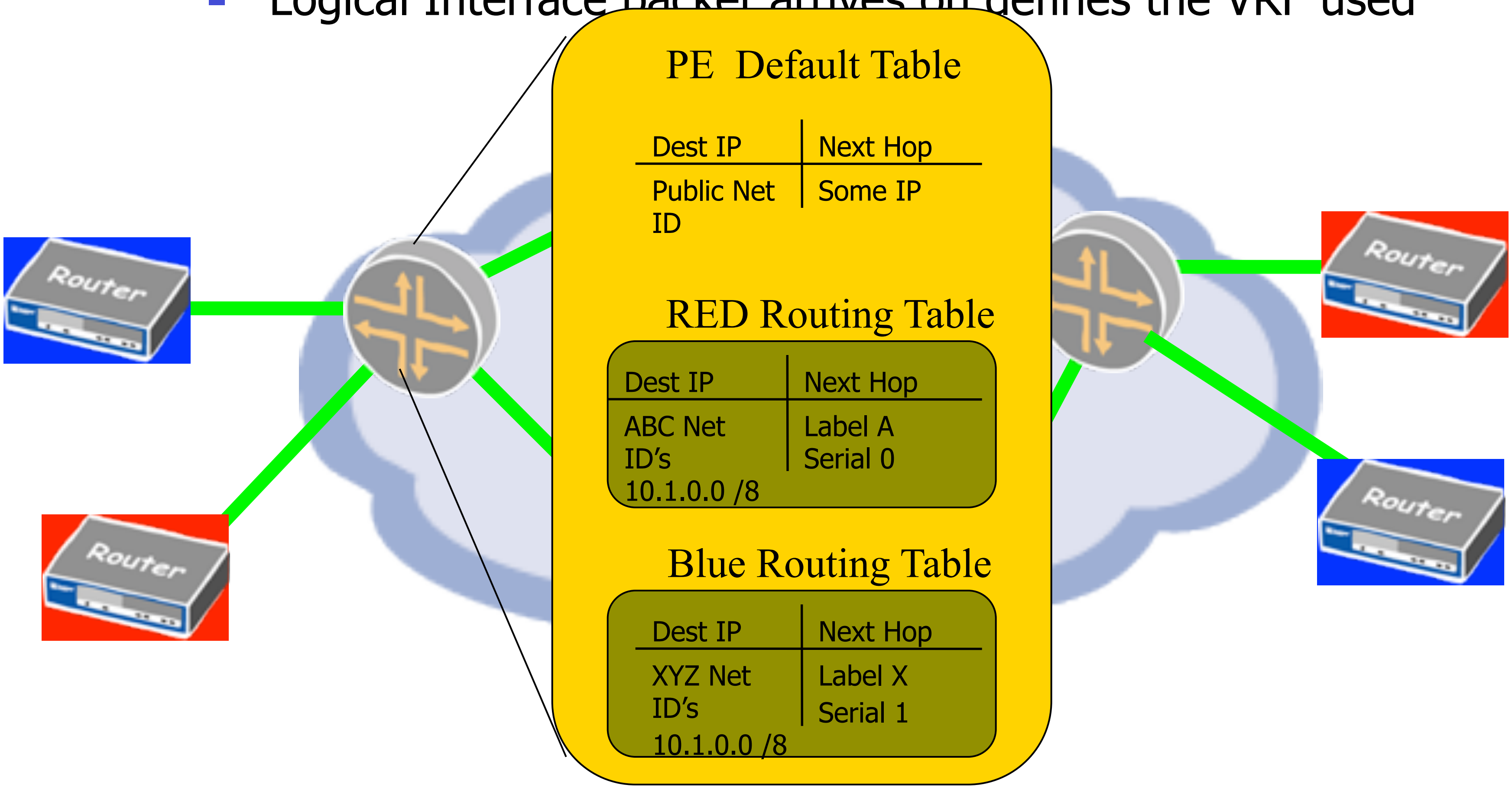
VRF – Virtual Routing and Forwarding instance

- VRF per VPN on PE
- Logical Interface packet arrives on defines the VRF used



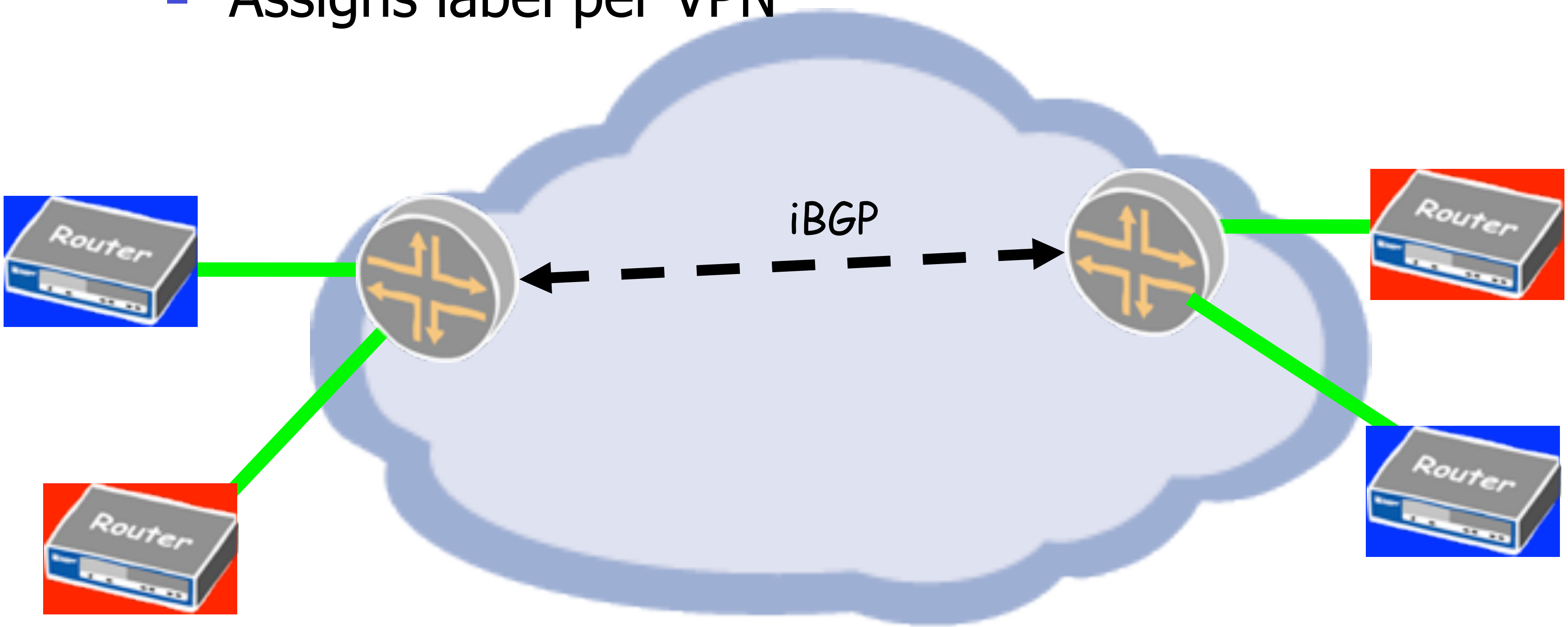
VRF – Virtual Routing and Forwarding instance

- VRF per VPN on PE
- Logical Interface packet arrives on defines the VRF used



PE-PE interaction

- iBGP between PE's carries routing information
- Assigns label per VPN



Route Distinguishers

- Used to disambiguate possibly duplicate routes from VRF’s
  - i.e. guarantee unique addressing space
  - AS:nn e.g. 100:23
  - IPv4:nn e.g. 192.168.1.1:23
- Creates a guaranteed unique address BGP can advertise in a single databas
- VPNIPv4 addresses

Route Distinguishers

RD's have nothing to do with defining VPN membership

- Used to disambiguate possibly duplicate routes from VRF's
    - i.e. guarantee unique addressing space
    - AS:nn e.g. 100:23
    - IPv4:nn e.g. 192.168.1.1:23
- Creates a guaranteed unique address BGP can advertise in a single databas
- VPNIPv4 addresses

RD's in action

- Per VPN via BGP label assignment
- PE – PE set up via LDP or RSVP (saves state)

RD's in action

- Per VPN via BGP label assignment
- PE – PE set up via LDP or RSVP (saves state)

Route Targets

- PE receives VPN IPv4 NLRI's
- Routes then placed into VRF based up RT
  - Extended BGP community,
  - AS:nn 100:45
  - IPv4:nn e.g. 192.168.1.1:45
- A route may have one or more RT

Route Targets

- PE receives VPN IPv4 NLRI's
- Routes then placed into VRF based up RT
  - Extended BGP community,
  - AS:nn 100:45
  - IPv4:nn e.g. 192.168.1.1:45
- A route may have one or more RT

Route Targets in action

- When routes are advertised, they are exported with one or more RT's
- A VRF can import routes with matching RT's
- Security of this architecture depends on YOUR provisioning integrity

Why RD's and RT's?

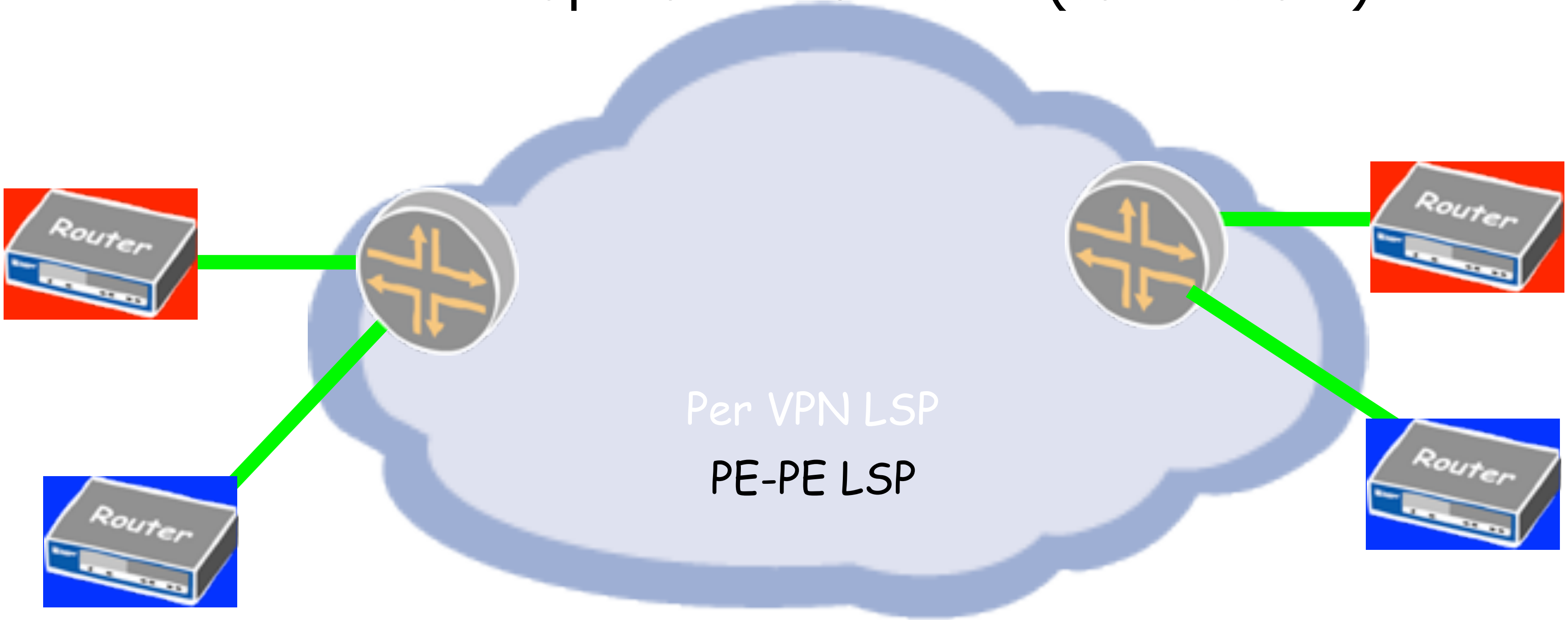
- Overhead is better when
  - Advertisements get bigger, as opposed t
  - More advertisements
- Allows for overlapping VPN's
- Can be the same
  - But don't lock yourself in

Why RD's and RT's?

- Overhead is better when
  - Advertisements get bigger, as opposed t
  - More advertisements
- Allows for overlapping VPN's
- Can be the same
  - But don't lock yourself in

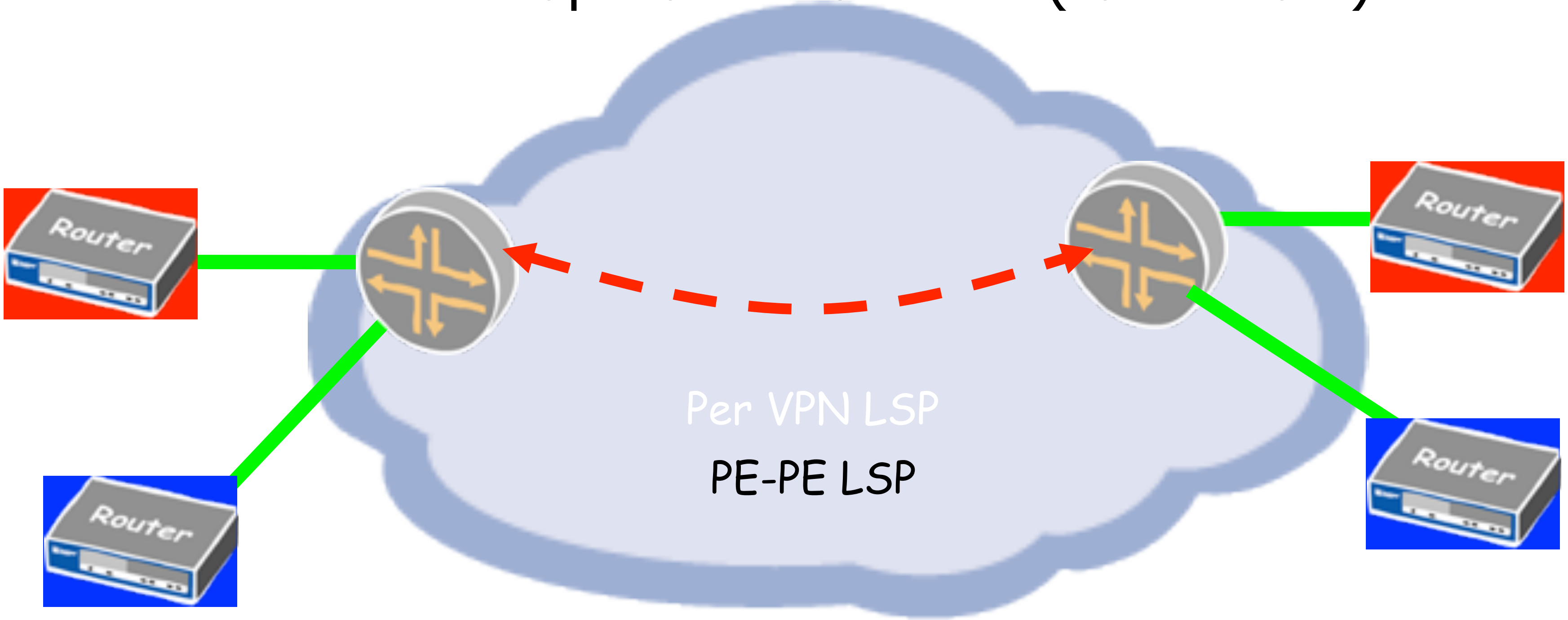
LSP establishment

- Per VPN via BGP label assignment
- PE – PE set up via LDP or RSVP (saves state)



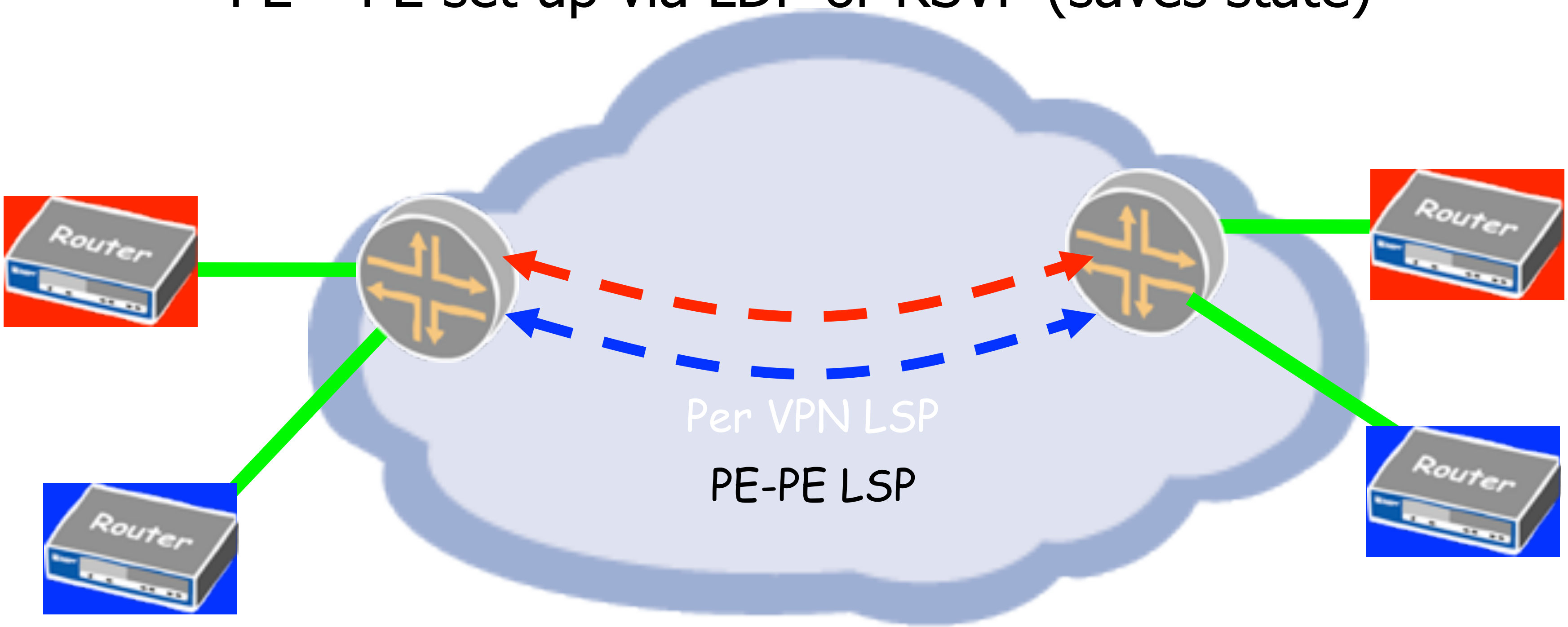
LSP establishment

- Per VPN via BGP label assignment
- PE – PE set up via LDP or RSVP (saves state)



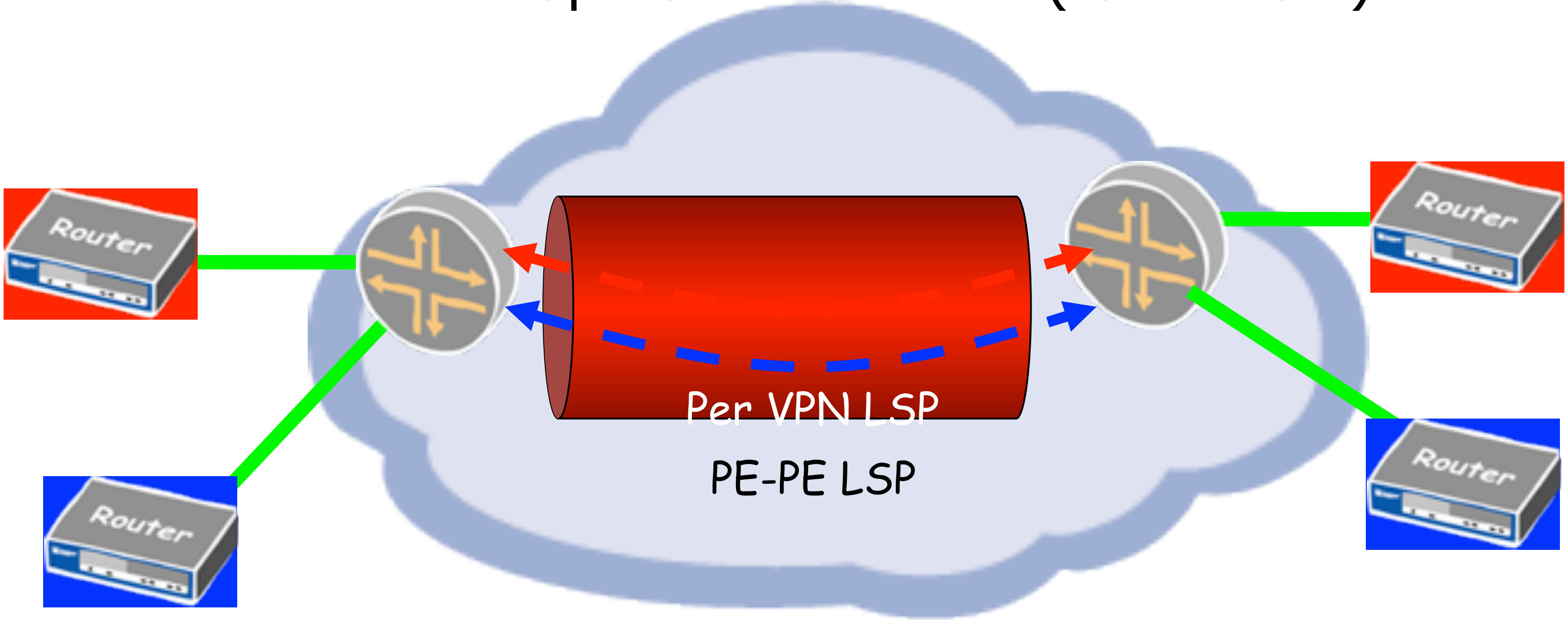
LSP establishment

- Per VPN via BGP label assignment
- PE – PE set up via LDP or RSVP (saves state)



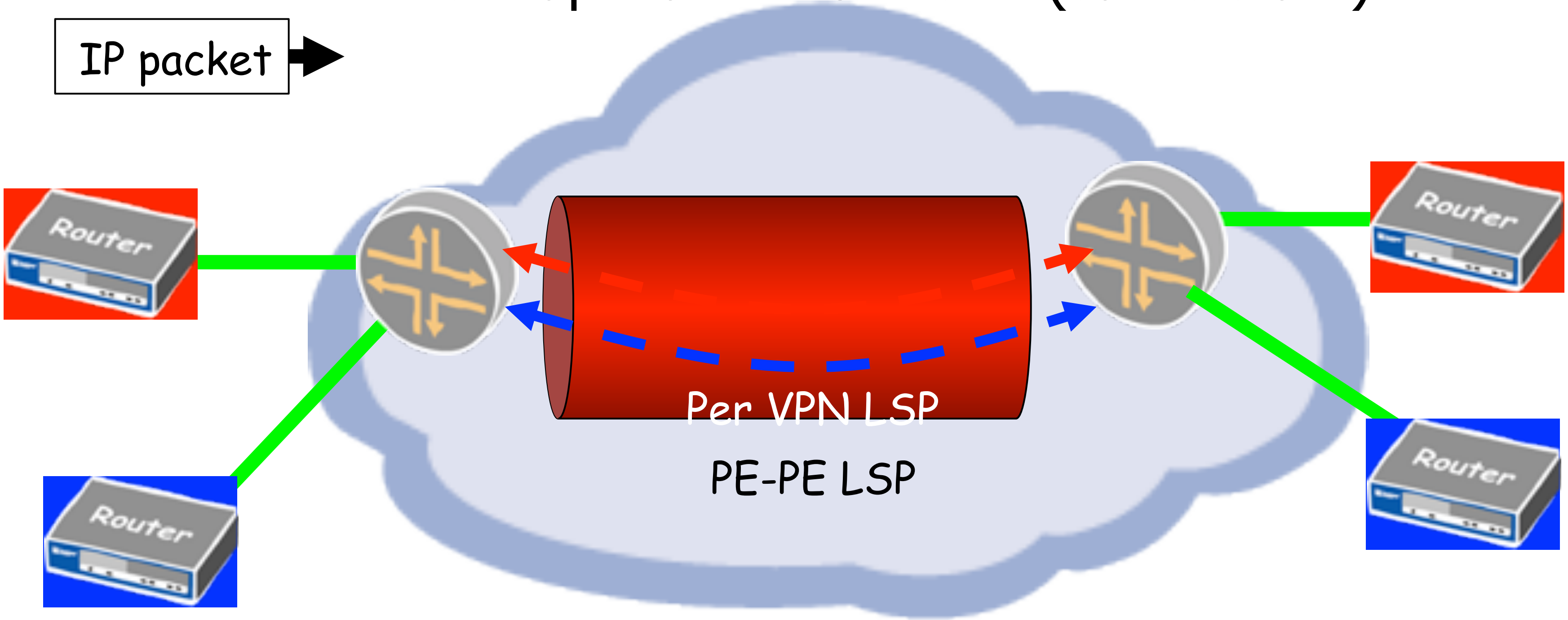
LSP establishment

- Per VPN via BGP label assignment
- PE – PE set up via LDP or RSVP (saves state)



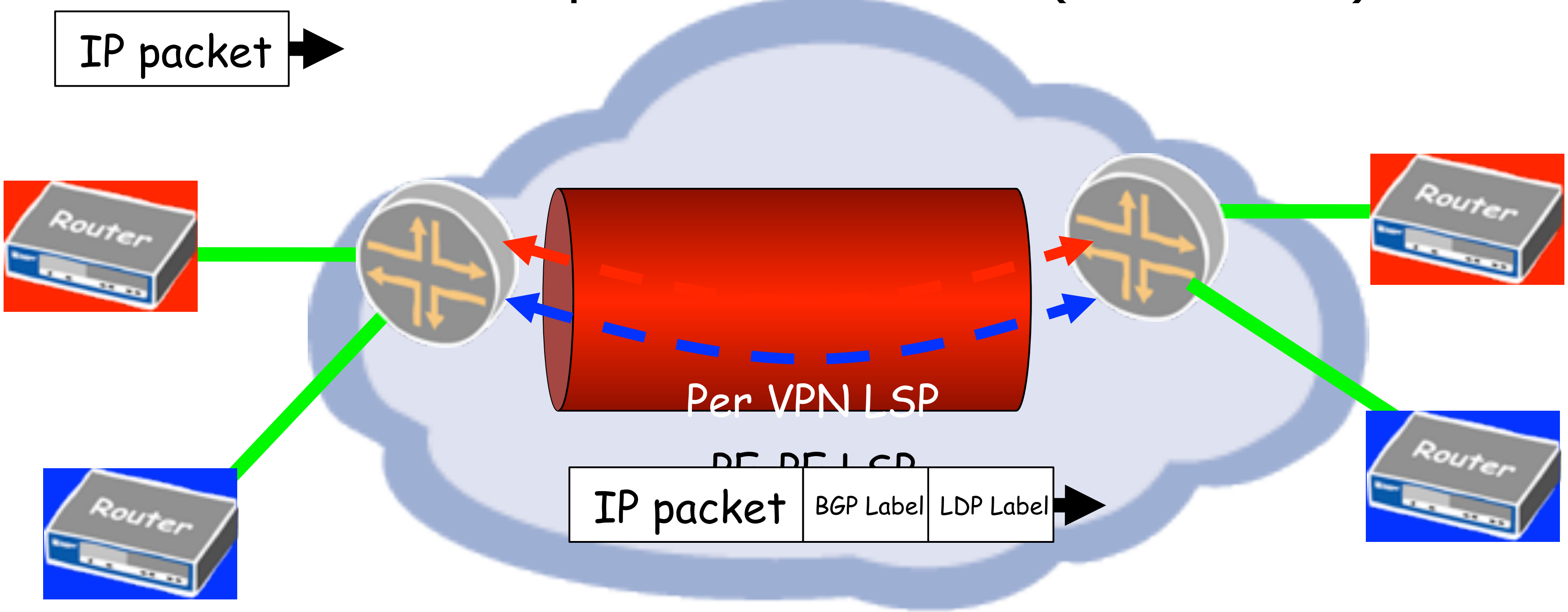
LSP establishment

- Per VPN via BGP label assignment
- PE – PE set up via LDP or RSVP (saves state)



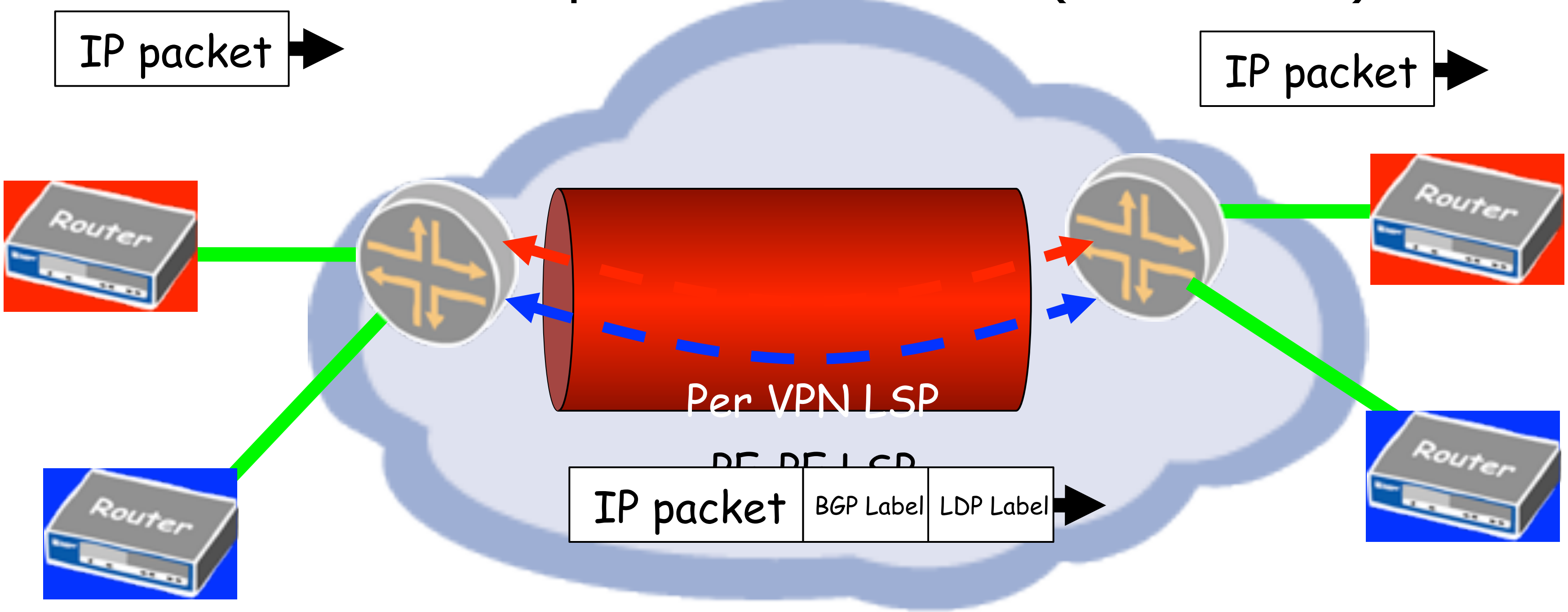
LSP establishment

- Per VPN via BGP label assignment
- PE – PE set up via LDP or RSVP (saves state)



LSP establishment

- Per VPN via BGP label assignment
- PE – PE set up via LDP or RSVP (saves state)

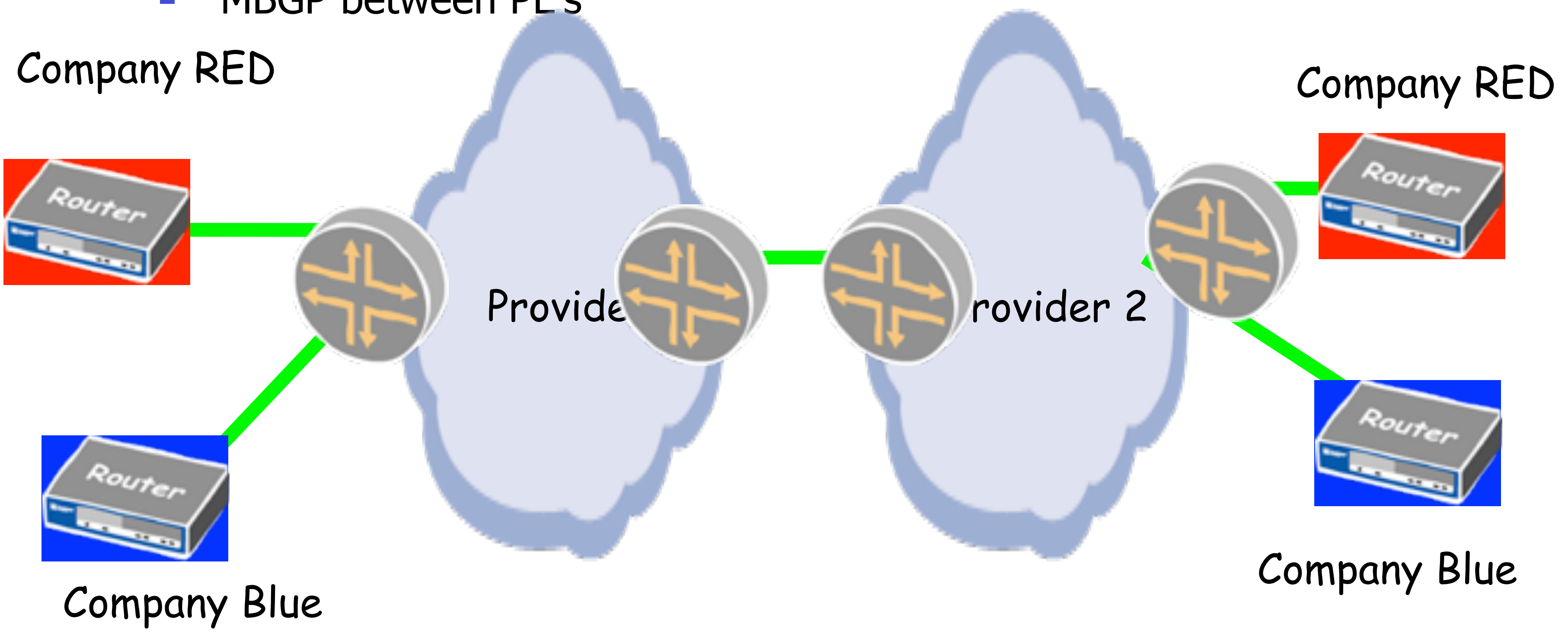


Connectivity

- Hub and spoke
  - Outsourcing internet access and Applications
- Full Mesh

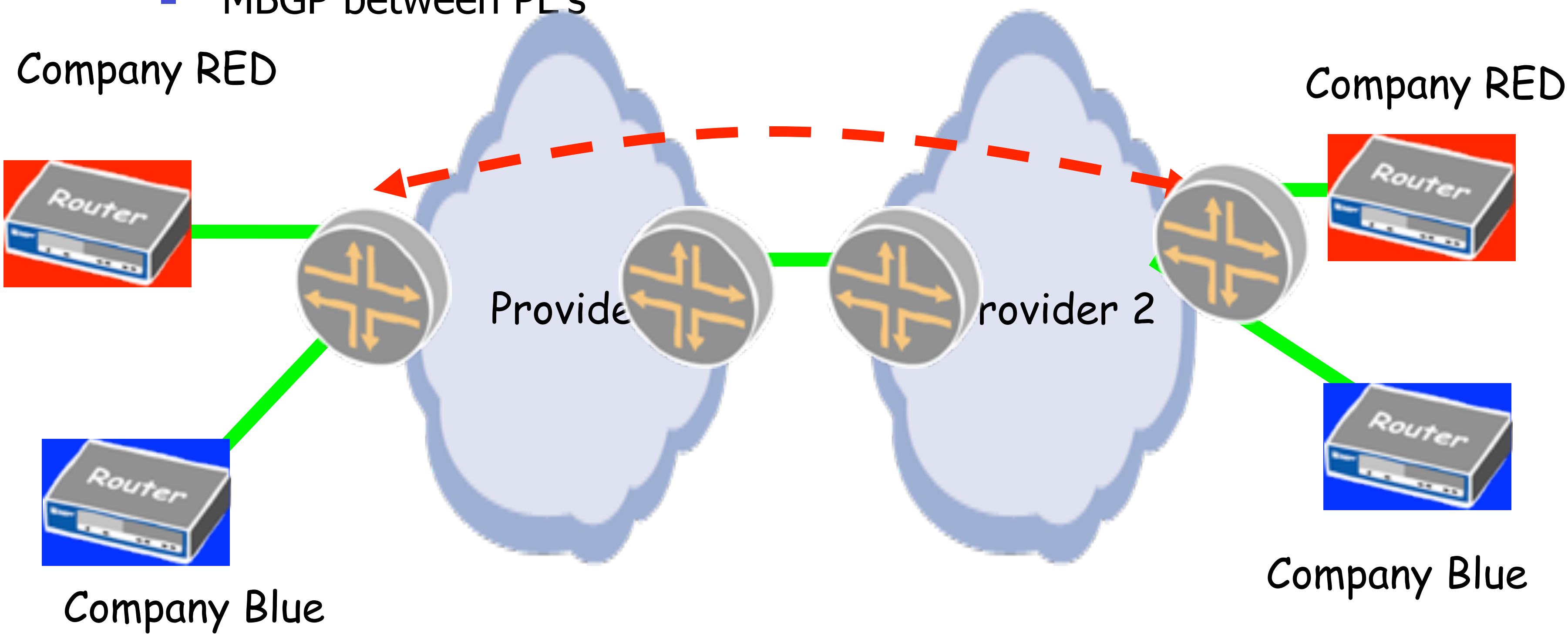
InterAS VPN's

- VRF-to-VRF
- MBGP between ASBR (not OSPF)
- MBGP between PE's



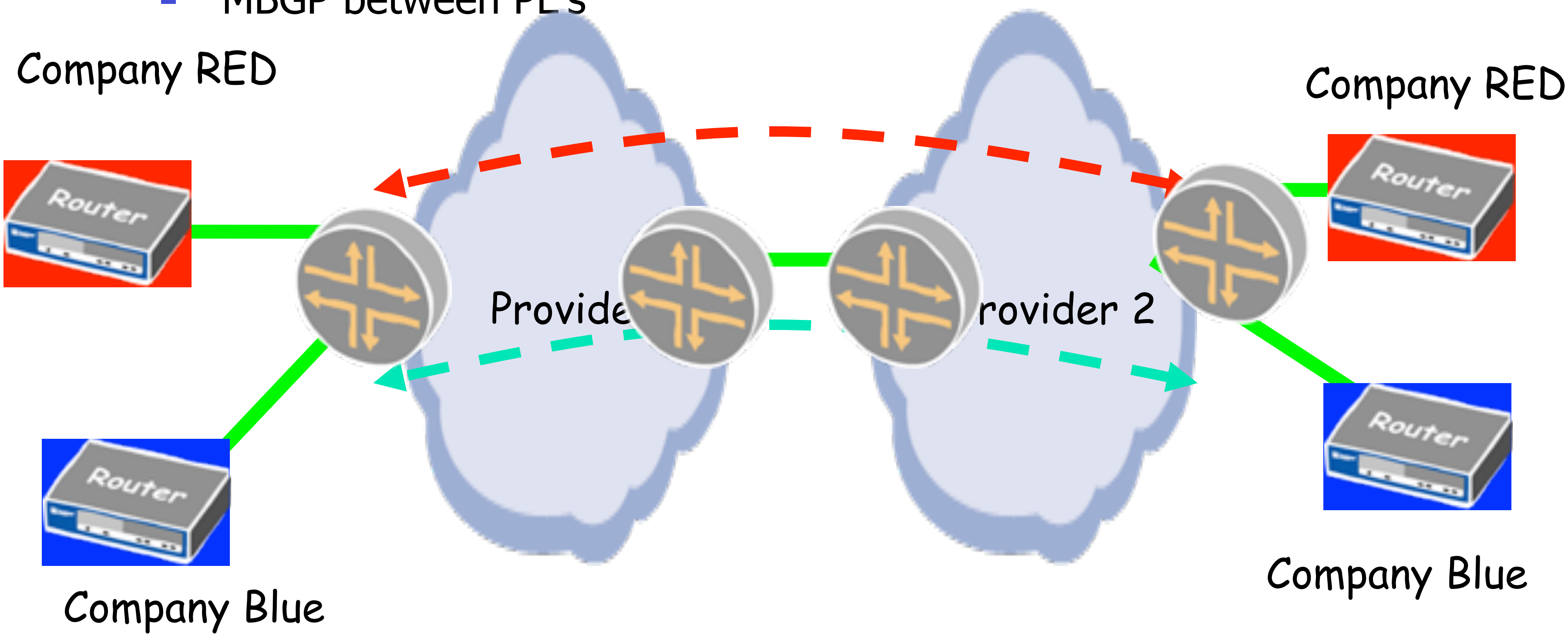
InterAS VPN's

- VRF-to-VRF
- MBGP between ASBR (not OSPF)
- MBGP between PE's



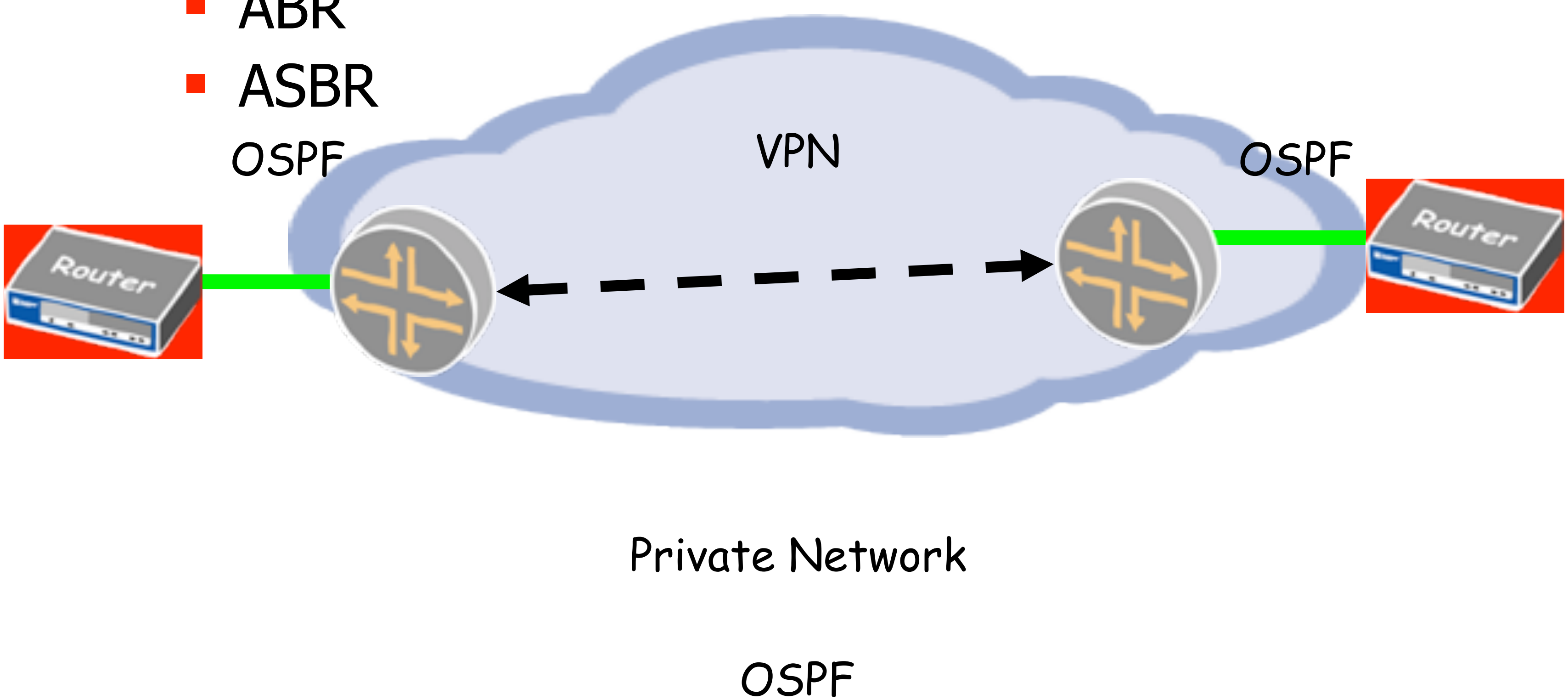
InterAS VPN's

- VRF-to-VRF
- MBGP between ASBR (not OSPF)
- MBGP between PE's



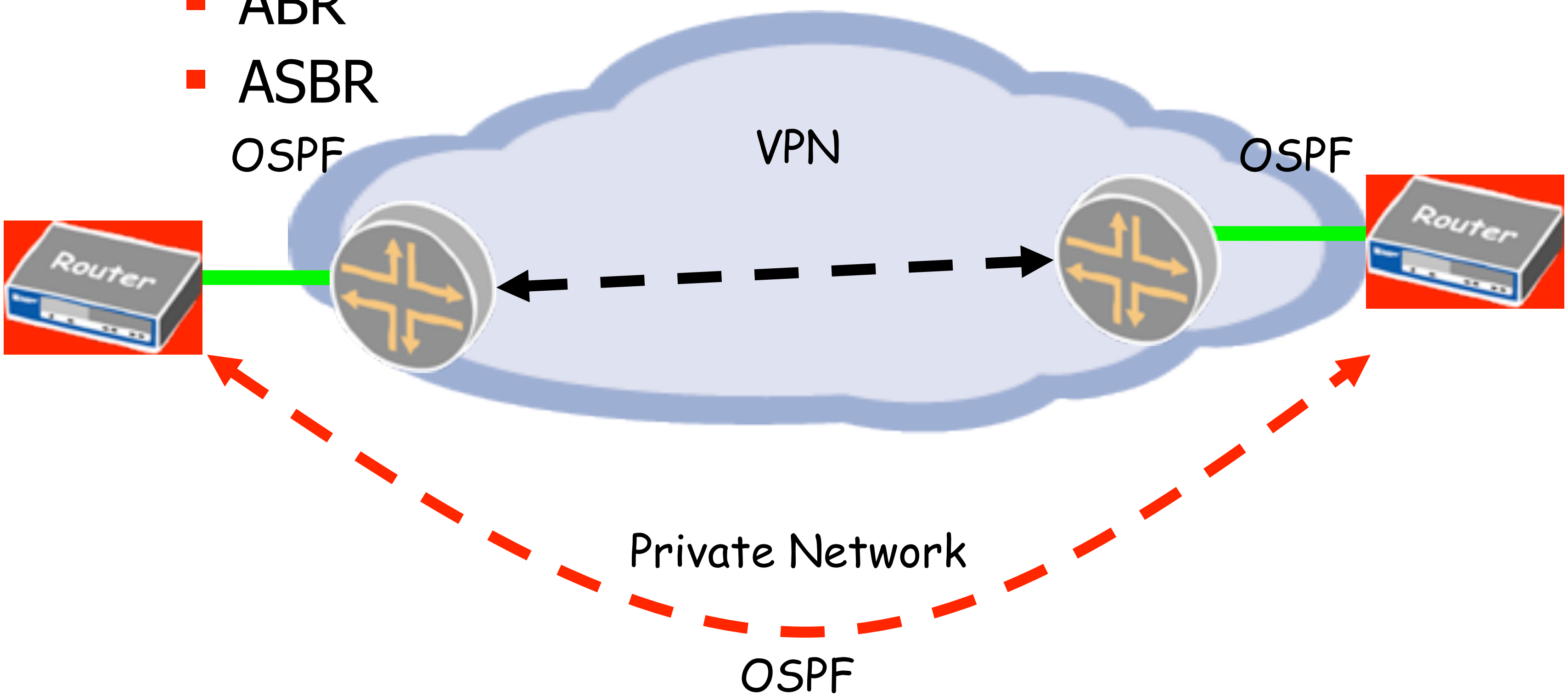
VPN as backup

- Do you want PE to appear as
  - Intra Area Router (Sham Links)
  - ABR
  - ASBR



VPN as backup

- Do you want PE to appear as
  - Intra Area Router (Sham Links)
  - ABR
  - ASBR



Issues

- BGP scaling
  - RR, often separate from IP RR
- Inter-AS scaling
  - MBGP between PE's is desirable
- Management
  - Usual MPLS, OAM, root cause automation.
  - Overlap NOC with VPN? Addressing?
- QoS
  - Carriers mapping 4+ queues

Configuring L3VPN's

Enable MPLS and LDP

```
JUNOS
-----
protocols {
    mpls {
        interface all;
    }
}
protocols {
    ldp {
        interface all;
    }
}
interfaces {
    fe-0/0/1 {
        unit 0 {
            family mpls;
        }
    }
}
```

```
IOS
-----
ip cef
mpls ip
mpls label protocol ldp
!
interface fast 0/1
mpls ip
mpls label protocol ldp

!
```

PE-PE MP-IBGP Peering

- PE-to-PE MP-IBGP sessions require VPN-IPv4 NLRI

```
JUNOS
-----
group int {
    type internal;
    local-address 192.168.24.1;
    family inet {
        unicast;
    }
    family inet-vpn {
        unicast;
    }
    neighbor 192.168.16.1;
}
```

```
IOS
-----
router bgp 150
neighbor 192.168.16.1 activate
!
address-family vpnv4
neighbor 192.168.16.1 activate
neighbor 192.168.16.1 send-community
extended
```

MP-IBGP Peering: PE-PE

```
lab@Amsterdam> show bgp neighbor
Peer: 192.168.16.1+179 AS 65412 Local: 192.168.24.1+1048 AS 65412
  Type: Internal    State: Established    Flags: <>
  Last State: OpenConfirm   Last Event: RecvKeepAlive
  Last Error: None
  Options: <Preference LocalAddress HoldTime AddressFamily Rib-group Refresh>
  Address families configured: inet-unicast inet-vpn-unicast
  Local Address: 192.168.24.1 Holdtime: 90 Preference: 170
  Number of flaps: 0
  Peer ID: 192.168.16.1     Local ID: 192.168.24.1     Active Holdtime: 90
  Keepalive Interval: 30
  NLRI advertised by peer: inet-unicast inet-vpn-unicast
  NLRI for this session: inet-unicast inet-vpn-unicast
  Peer supports Refresh capability (2)
  Table inet.0 Bit: 10000
    Send state: in sync
    Active prefixes: 0
    Received prefixes: 0
    Suppressed due to damping: 0
  Table bgp.l3vpn.0 Bit: 30000
    Send state: in sync
    Active prefixes: 8
    Received prefixes: 8
    Suppressed due to damping: 0
  Table vpn-a.inet.0 Bit: 40000
    Send state: in sync
    Active prefixes: 7
    Received prefixes: 8
```

Assigning the Route Distinguisher

- Manually assign the RD per VRF table

```
JUNOS
-----
[edit routing-instances vpn-a]
lab@HK# show
instance-type vrf;
interface fe-0/0/0.0;
route-distinguisher 192.168.16.1:1;
…
```

```
IOS
---
ip vrf ODD_Customer
rd 150:101
…
```

- Enable router to dynamically assign a unique Type 1 RD to every configured VRF table

```
[edit routing-options]
lab@HK# show
…
route-distinguisher-id 192.168.16.1;
autonomous-system 65412;
```

A Sample VRF Table Configuration

Create a VRF table called *vpn-a* with BGP running between the PE and CE routers using the `vrf-target` statement:

```
JUNOS
-----
[edit routing-instances vpn-a]
lab@HK# show
Vrf-table-label; <------------------
instance-type vrf;
interface fe-0/0/0.0;
route-distinguisher 3:1;
vrf-target {
        import target:3:111;
        export target:3:111;
}
```

```
IOS
-----
ip vrf vpn-a
rd 3:101

interface fastethernet 0/0
ip vrf forwarding vpn-a
ip address 200.1.9.1 255.255.255.0


ip vrf vpn-a
route-target export 3:111
route-target import 3:111
```

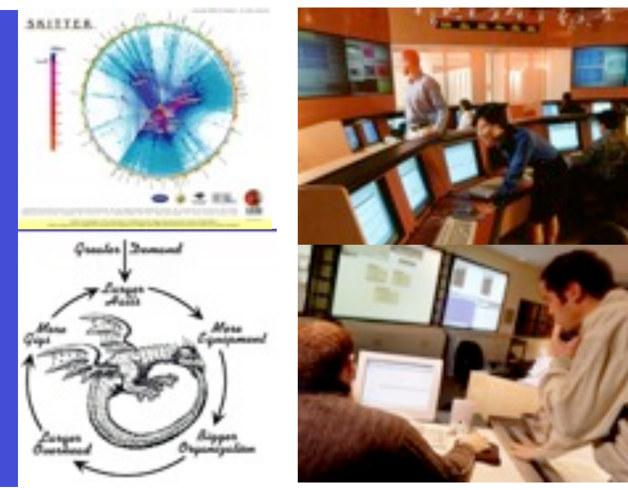
Further Reading

1. http://www.juniper.net/solutions/literature/white_papers/
2. http://www.juniper.net/solutions/literature/white_papers/200012.pdf
3. www.mplsrc.com

MPLS / L3 VPN Security

- RFC2547bis
- BGP / IP VPN
- Other Options
  - Virtual / logical routers – simpler to understand perhaps, but scaling issues

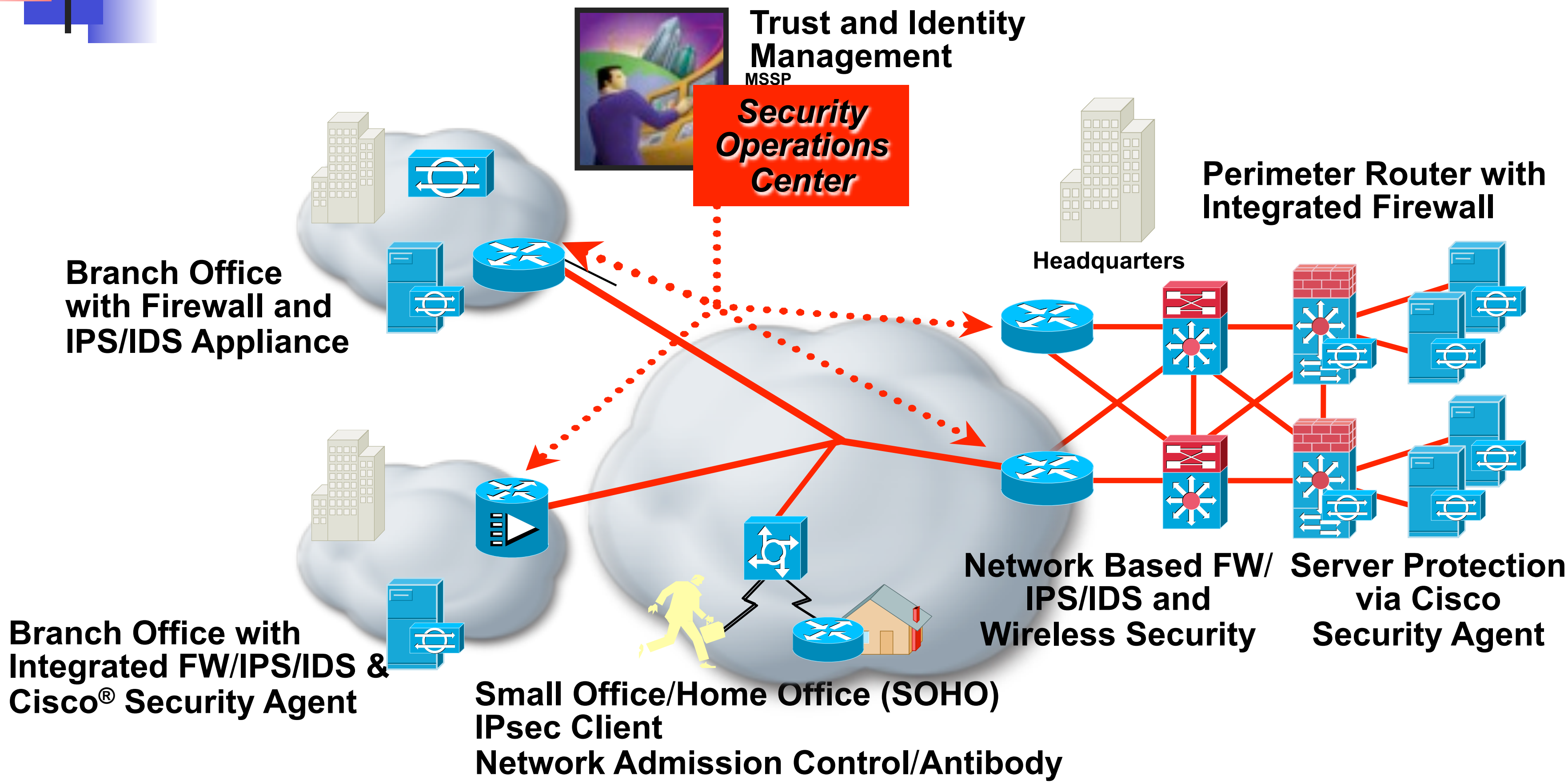
Understanding IPSec and SSL VPN



Managed Security Services

- Managed Security Services Architecture
- Managed IP VPN
- Managed IPSec VPN

Managed Security Services Architecture

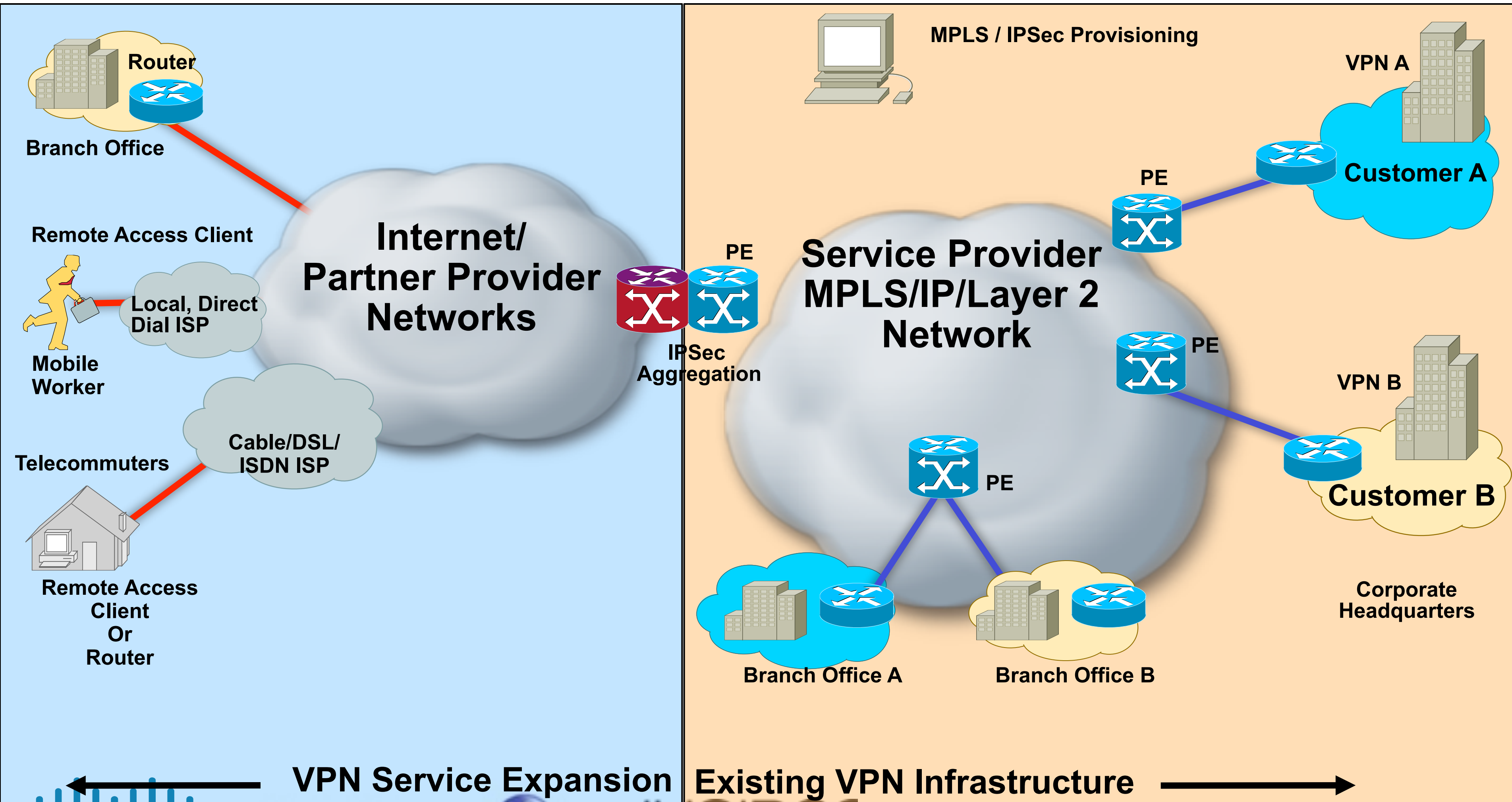


Managed Security Services

End-to-end security with integrated firewalls and intrusion prevention / detection systems

Secure data/operations center with integrated services

Managed IP VPN Security Services



Applying Cryptography to IP VPN

- IP Security [IPSec]
- Secure Socket Layer /Transport Layer Security

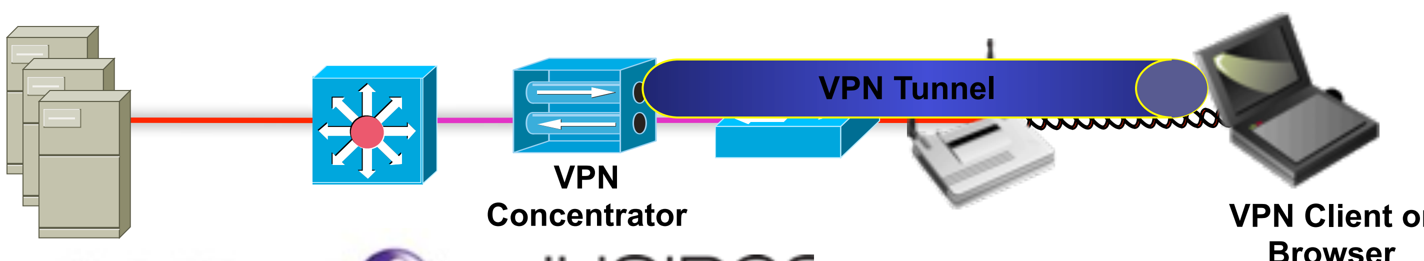
- Applying cryptography helps integrate:
  - Availability
  - Integrity
  - Confidentiality
  - Anti-replay

- That means using firewalls for access control and using SSL/TLS & IPsec for confidentiality and data origin authentication.

Virtual Private Network (VPN) Overview

IP Security (IPSec) and SSL

- Mechanism for secure communication over IP
  - Authenticity (Unforged/trusted party)
  - Integrity (Unaltered/tampered)
  - Confidentiality (Unread)
- Remote Access (RA) VPN Components
  - Client (mobile or fixed)
  - Termination device (high number of endpoints)

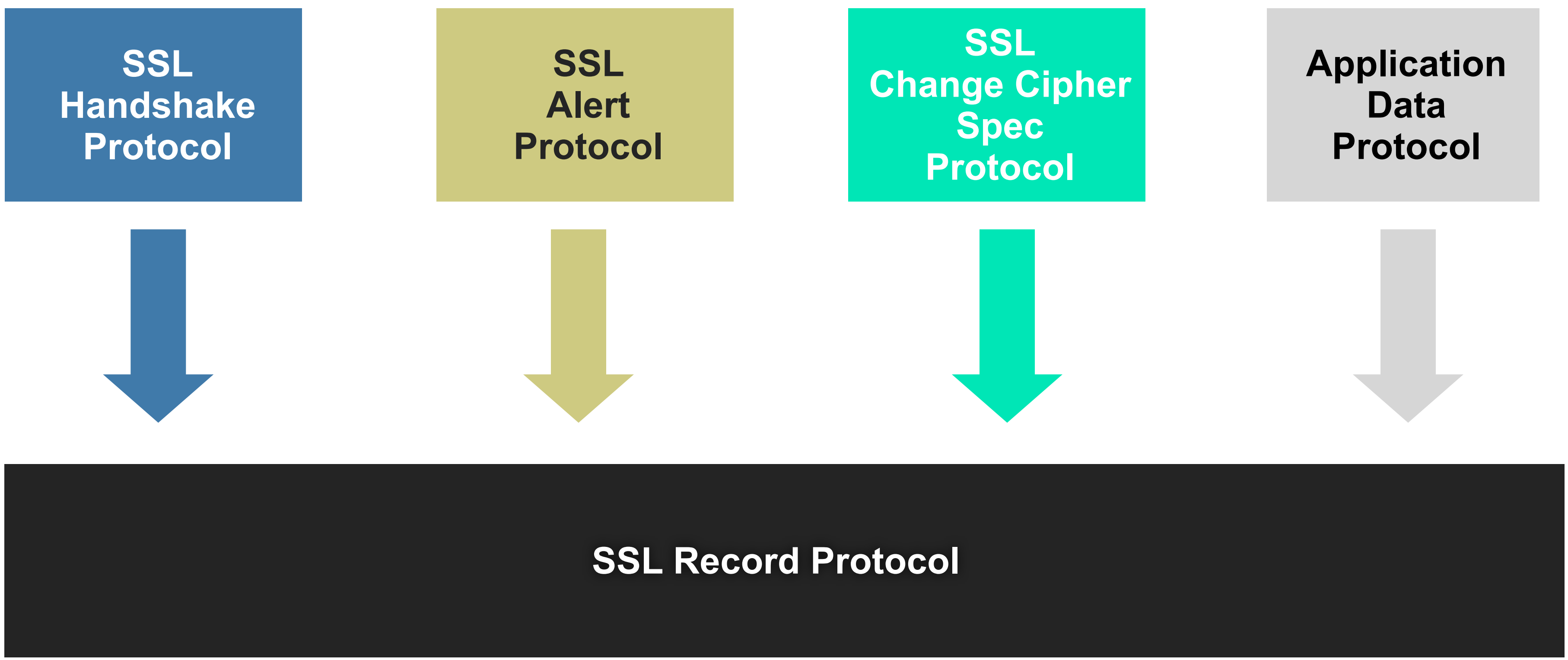


SSL/TLS

- SSL and TLS
  - SSL v3.0 specified in an I-D in 1996 (draft-freier-ssl-version3-02.txt)
  - TLS v1.0 specified in RFC 2246 in 1999
  - TLS v1.0 = SSL v3.1 ≈ SSL v3.0
- OSI layer placement
  - Above TCP/IP and below application layer
  - Most common use with HTTP → HTTPS
- Goals of protocol
  - Secure communication between applications
  - Authentication + privacy + integrity

SSL Composition

SSL Is a Combination of a Primary Record Protocol with Four 'Client' Protocols



SSL Protocols

SSL Handshake Protocol	Allow for Authentication and Generation of Encryption Material Through Negotiation of Parameters And Exchange of Calculated Values
SSL Alert Protocol	Used to Convey Administrative Alerts for Managing SSL Connections and Sessions
SSL Change Cipher Spec Protocol	Used to Signal Transition to New Cipher and Keys Generally Towards the End of a Handshake Negotiation
SSL Record Protocol	Provides for Transmission of Data in Encrypted and Compressed Form with Integrity Checking

How Does SSL Negotiation Work?

SSL Session Is Negotiated Through Four Sets Of Messages

1st Set of Messages

Used to Start a Negotiation and to Offer and Agree upon Basic Negotiation Options

2nd Set of Messages

Used by the Server to Prove Its ID to the Client and to Send Its Certificate

3rd Set of Messages

Optionally Used By the Client to Prove Its ID and to Send Its Certificate, if Needed, and to Pass Initial Keying Material for Subsequent Key Generation to the Server

4th Set of Messages

Used by the Server and Client to Indicate Beginning of Use of New Keying Material

SSL/TLS Properties

- Connection is private
  - Encryption is used after an initial handshake to define a secret key.
  - Symmetric cryptography used for data encryption (DES or RC4).
- Peer's identity can be authenticated
  - Asymmetric cryptography is used (RSA or DSS).
- Connection is reliable
  - Message transport includes a message integrity check using a keyed MAC.
  - Secure hash functions (such as SHA and MD5) are used for MAC computations.

SSL Handshake Process

SSL Handshake Process

SSL Protocol Elements

- Handshake Protocol
  - Negotiates crypto algorithms and keys
- Alert Protocol
  - Indicates errors or end of a session
- Record Protocol
    - Functions as layer beneath all SSL messages
    - Indicates which integrity and encryption protection is applied to data
  - Each record individually encrypted and hashed
  - Connections closed with a 'Close Notify'
  - Previously established session can be resumed by providing session ID in 'Client Hello'
    - Abbreviated version of handshake protocol
    - Reuses previously established crypto parameters

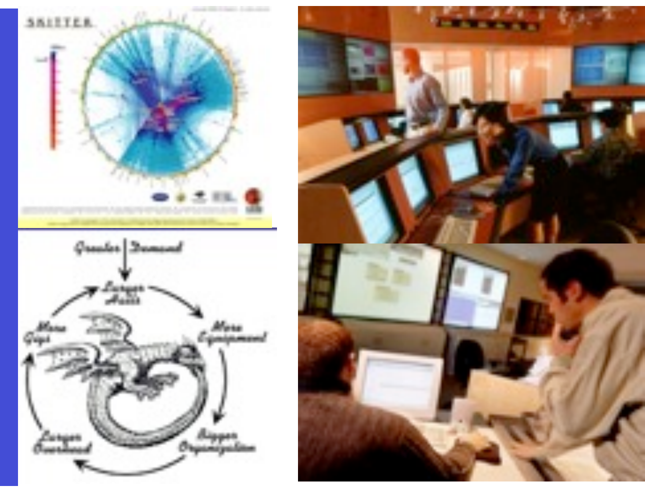
SSL Client Authentication

- Client authentication (certificate based) is optional and not often used
- Many application protocols incorporate their own client authentication mechanism such as username/password or S/Key
- These authentication mechanisms are more secure when run over SSL

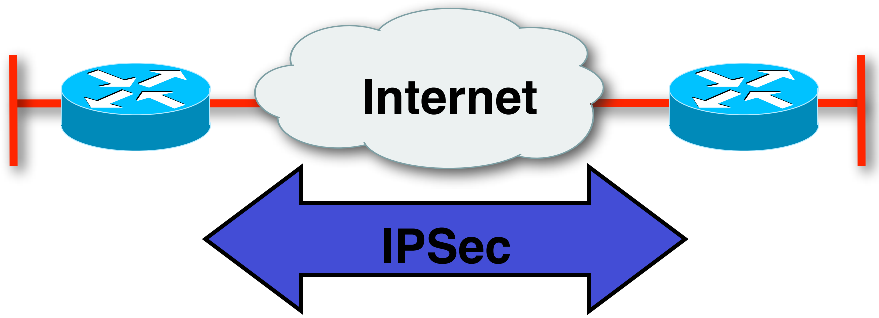
SSL / TLS Port Numbers

Protocol	Defined Port Number	SSL/TLS Port Number
HTTP	80	443
NNTP	119	563
SMTP	110	995
FTP-Data	20	989
FTP-Control	21	990
Telnet	23	992

IPSec Explained for MSS



What Is IPSec?



- IETF standard that enables encrypted communication between peers:
  - Consists of open standards for securing private communications
  - Network layer encryption ensuring data confidentiality, integrity, and authentication
  - Scales from small to very large networks
  - Available in Cisco IOS software version 11.3(T) and later
  - Included in PIX Firewall version 5.0 and later

IPSec Composition

IPSec Combines Three Main Protocols into a Cohesive Security Framework

IPSec Composition

IPSec Combines Three Main Protocols into a Cohesive Security Framework

IKE	Provides Framework for the Negotiation of Security Parameters and Establishment of Authenticated Keys

IPSec Composition

IPSec Combines Three Main Protocols into a Cohesive Security Framework

IKE	Provides Framework for the Negotiation of Security Parameters and Establishment of Authenticated Keys
ESP	Provides Framework for the Encrypting, Authenticating and Securing Data

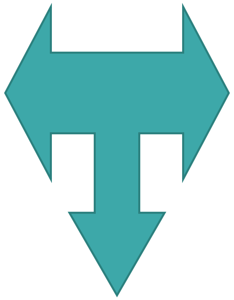
IPSec Composition

IPSec Combines Three Main Protocols into a Cohesive Security Framework

Protocol	Function
IKE	Provides Framework for the Negotiation of Security Parameters and Establishment of Authenticated Keys
ESP	Provides Framework for the Encrypting, Authenticating and Securing Data
AH	Provides Framework for the Authenticating and Securing Data

What Is IKE?

IKE (Internet Key Exchange) (RFC 2409)
Is a Hybrid Protocol



What Is IKE?

IKE (Internet Key Exchange) (RFC 2409)
Is a Hybrid Protocol

SKEME

Mechanism for Utilizing Public Key Encryption for Authentication

What Is IKE?

IKE (Internet Key Exchange) (RFC 2409)
Is a Hybrid Protocol

SKEME

Mechanism for Utilizing Public Key Encryption for Authentication

Oakley

Modes-Based Mechanism for Arriving at an Encryption Key between Two Peers

What Is IKE?

IKE (Internet Key Exchange) (RFC 2409) Is a Hybrid Protocol

SKEME

Mechanism for Utilizing Public Key Encryption for Authentication

Oakley

Modes-Based Mechanism for Arriving at an Encryption Key between Two Peers

ISAKMP

Architecture for Message Exchange Including Packet Formats and State Transitions between Two Peers

Why IKE?

IKE Solves the Problems of Manual and Unscalable Implementation of IPSec by Automating the Entire Key Exchange Process

- Negotiation of SA characteristics
- Automatic key generation
- Automatic key refresh
- Manageable manual configuration

How Does IKE Work?

Phase 1 Exchange

Peers Negotiate a Secure, Authenticated Channel with Which to Communicate 'Main Mode'
or 'Aggressive Mode' Accomplish a Phase I Exchange

How Does IKE Work?

IKE Is a TWO Phase Protocol

Phase 1 Exchange

Peers Negotiate a Secure, Authenticated Channel with Which to Communicate 'Main Mode'
or 'Aggressive Mode' Accomplish a Phase I Exchange

How Does IKE Work?

IKE Is a TWO Phase Protocol

Phase 1 Exchange

Peers Negotiate a Secure, Authenticated Channel with Which to Communicate 'Main Mode'
or 'Aggressive Mode' Accomplish a Phase I Exchange

How Does IKE Work?

IKE Is a TWO Phase Protocol

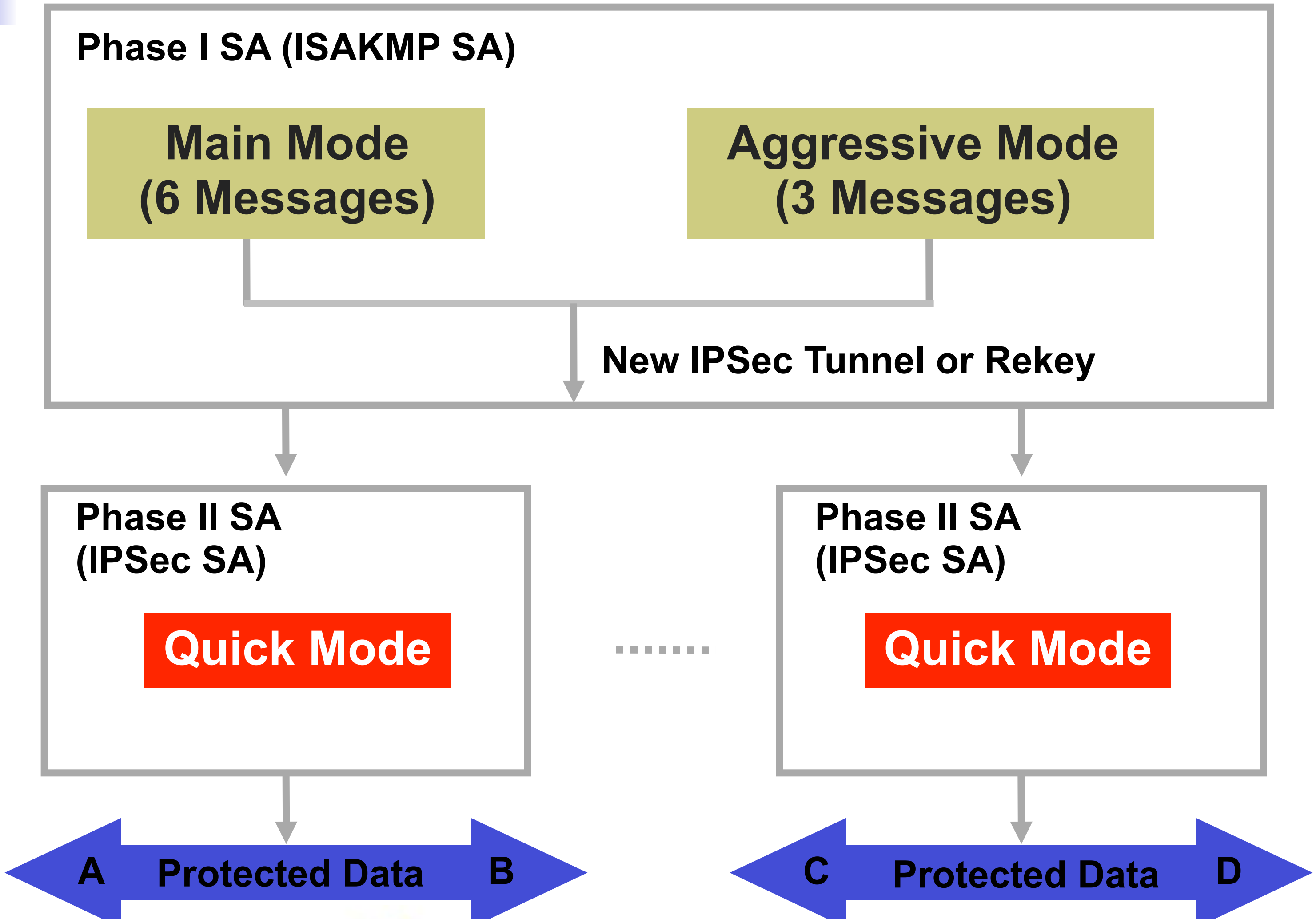
Phase 1 Exchange

Peers Negotiate a Secure, Authenticated Channel with Which to Communicate 'Main Mode' or 'Aggressive Mode' Accomplish a Phase I Exchange

Phase 2 Exchange

Security Associations Are Negotiated on Behalf of IPSec Services; 'Quick Mode' Accomplishes a Phase II Exchange

How Does IKE Work?



IKE v2: Replacement for Current IKE Specification

- Feature preservation
  - Most of the features and characteristics of the baseline parent IKE v1 protocol are being preserved in v2
- Compilation of features and extensions
  - Quite a few features that were added on top of the baseline IKE protocol functionality in v1 are being reconciled into the mainline v2 framework
- New features
  - A few new mechanisms and features are being introduced in the IKE v2 protocol as well

IKE v2: What Is Not Changing

- Features in v1 that have been debated but are ultimately being preserved in v2
  - Most payloads reused
  - Use of nonces to ensure uniqueness of keys
- v1 extensions and enhancements being merged into mainline v2 specification
  - Use of a 'configuration payload' similar to MODECFG for address assignment
  - 'X-auth' type functionality retained through EAP
  - Use of NAT Discovery and NAT Traversal techniques

IKE v2: What Is Changing

Significant Changes Being Made to the Baseline Functionality of IKE

- EAP adopted as the method to provide legacy authentication integration with IKE
- Public signature keys and pre-shared keys, the only methods of IKE authentication
- Use of 'stateless cookie' to avoid certain types of DOS attacks on IKE
- Continuous phase of negotiation

How Does IKE v2 Work?

IKE_SA_INIT
(Two Messages)

IKE_SA Authentication Parameters Negotiated

IKE_AUTH
(Two Messages)

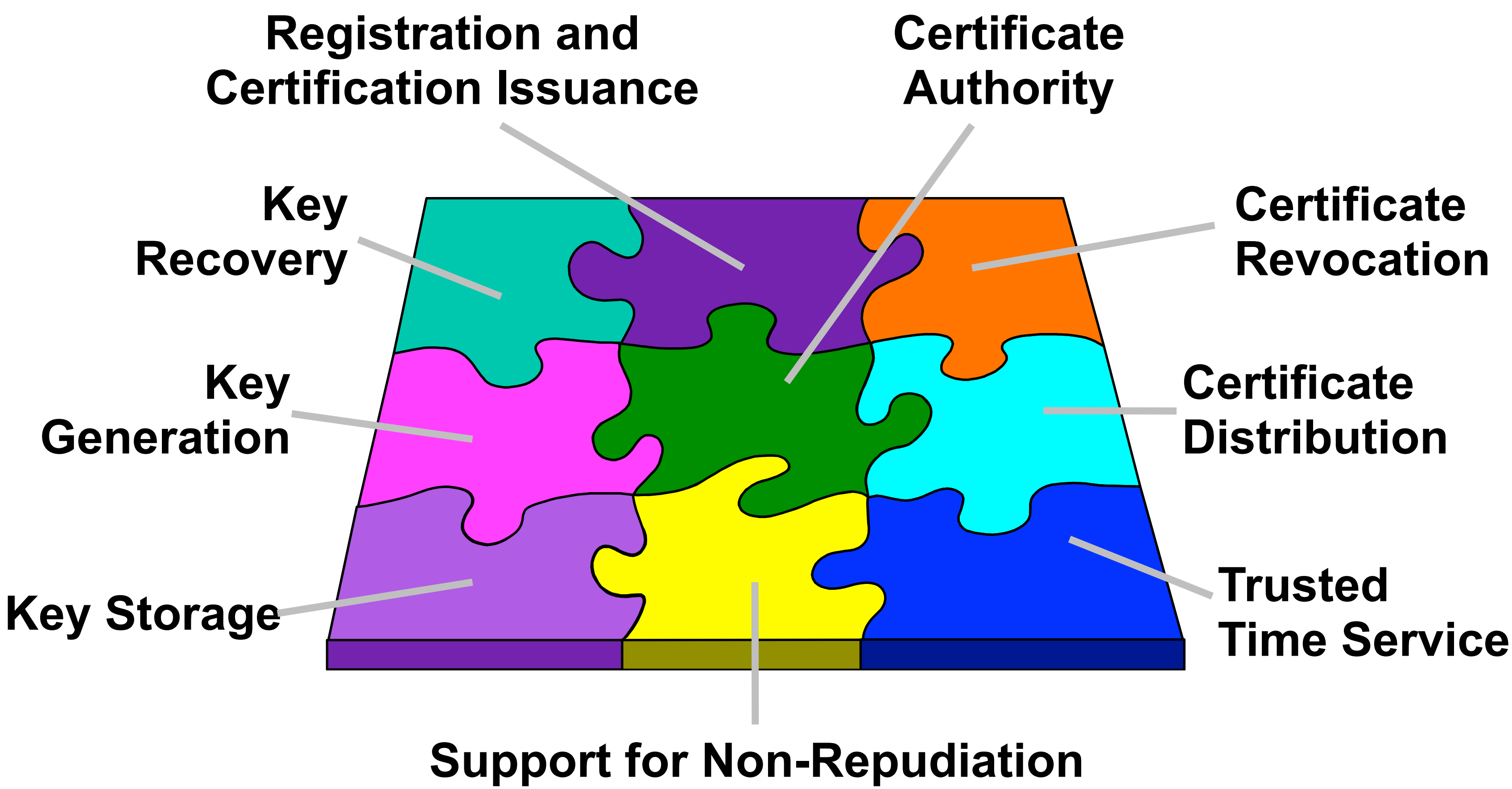
IKE Authentication Occurs and One CHILD_SA Created

CREATE_CHILD_SA
(Two Messages)

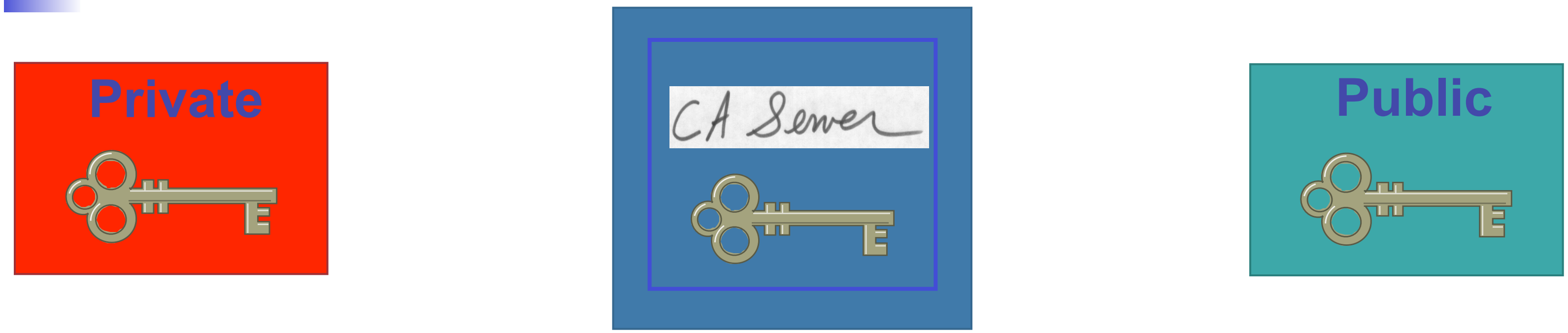
Second CHILD_SA Created

A Protected Data B

PKI: IKE Authentication Architecture



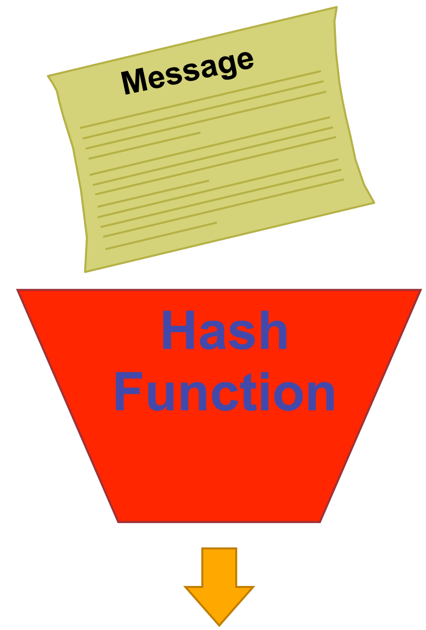
Digital Signatures



- Entity authentication
- Data origin authentication
- Integrity
- Non-repudiation

Digital Signatures

- One-way function; easy to produce hash from message, "impossible" to produce message from hash



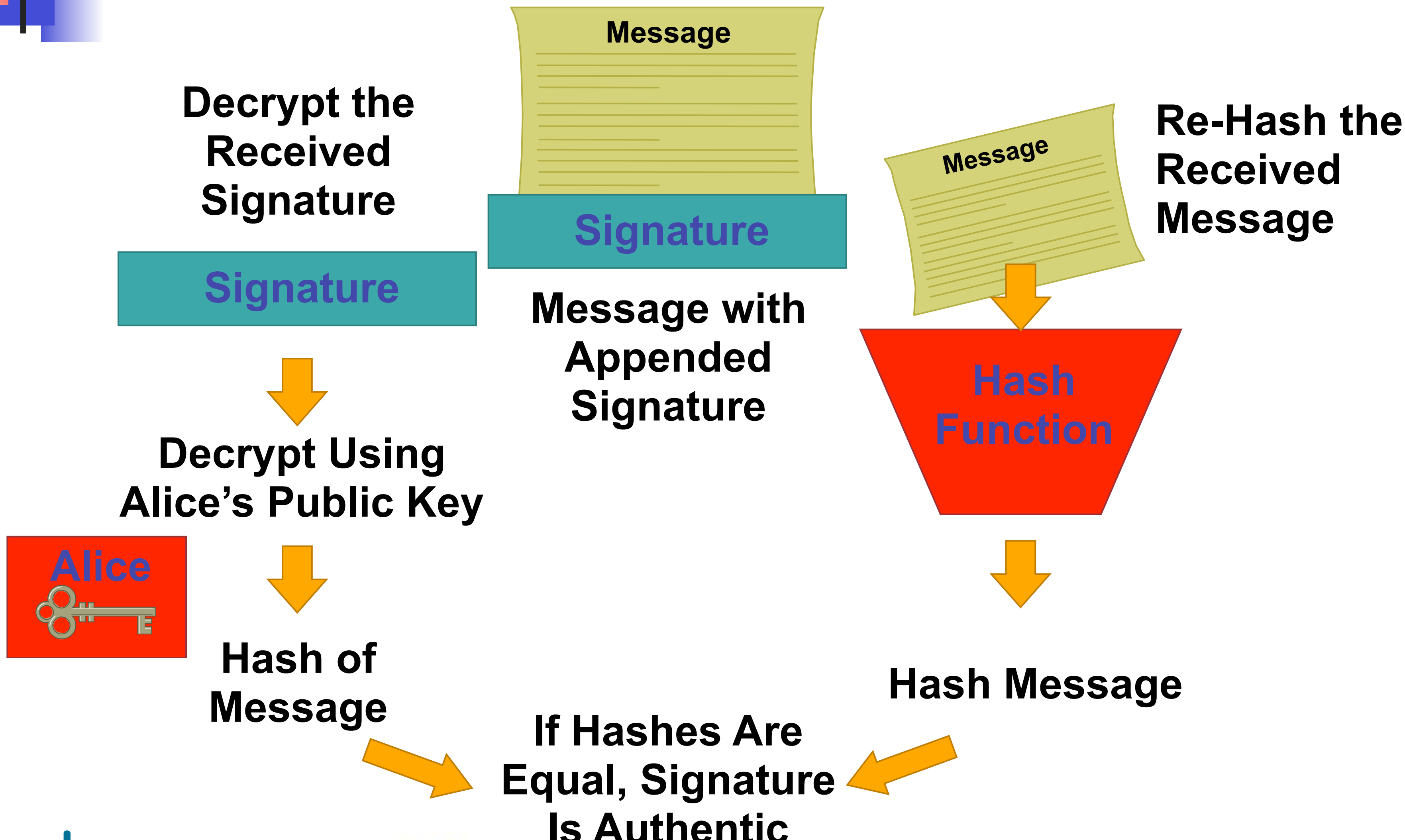
Hash of Message **s74hr7sh7040236fw**

Sign Hash with Private Key ***7sr7ewq7ytoj56o457***

Alice

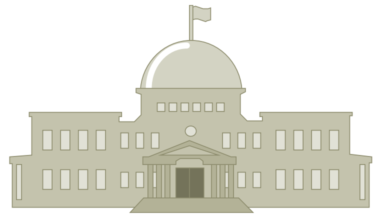
Signature = "Encrypted" Hash of Message

Signature Verification



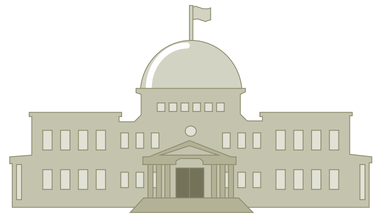
Digital Certification

Certificate Authority



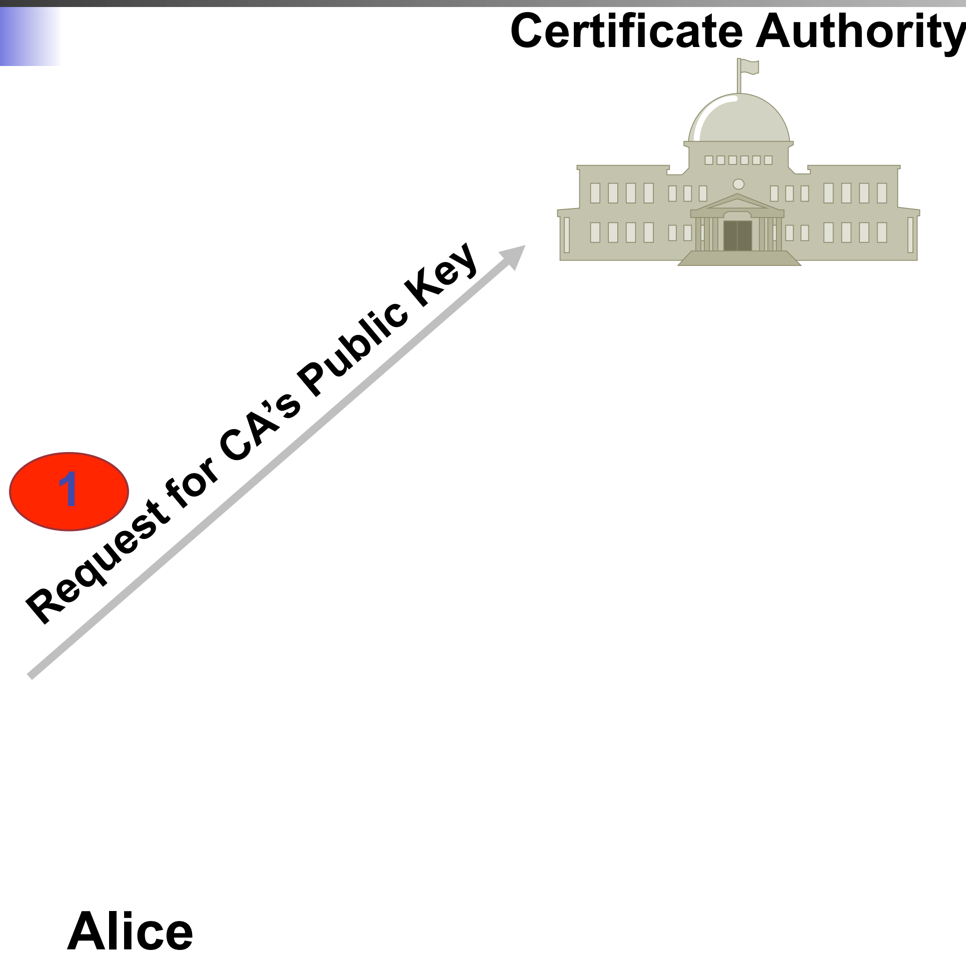
Digital Certification

Certificate Authority



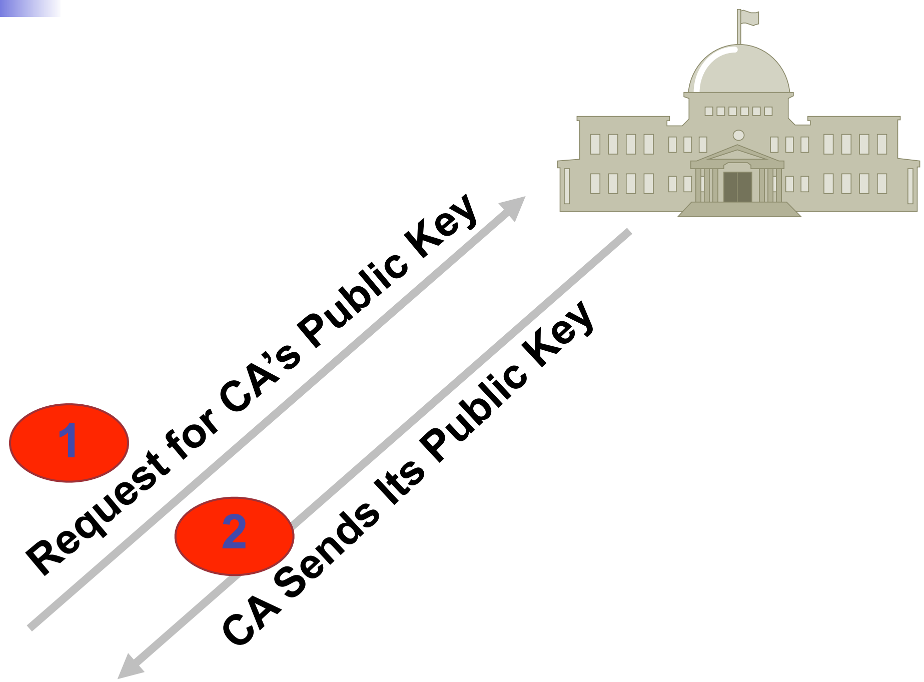
Alice

Digital Certification



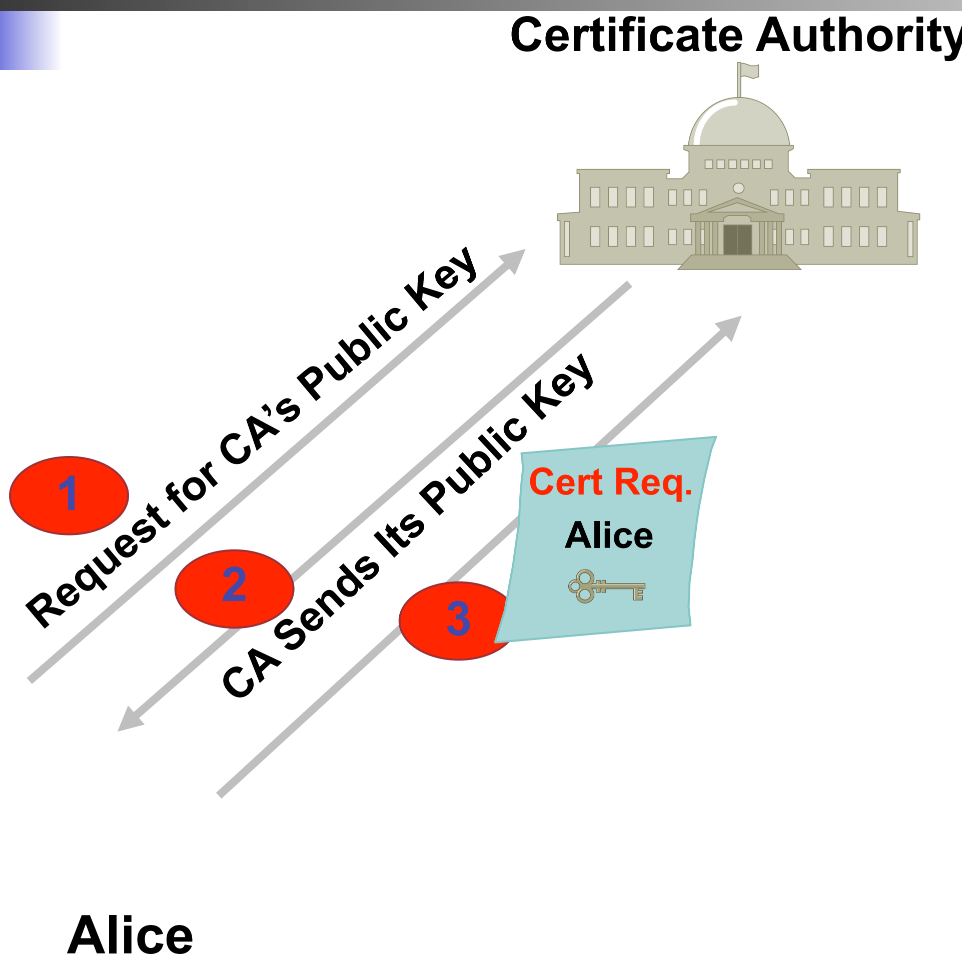
Digital Certification

Certificate Authority

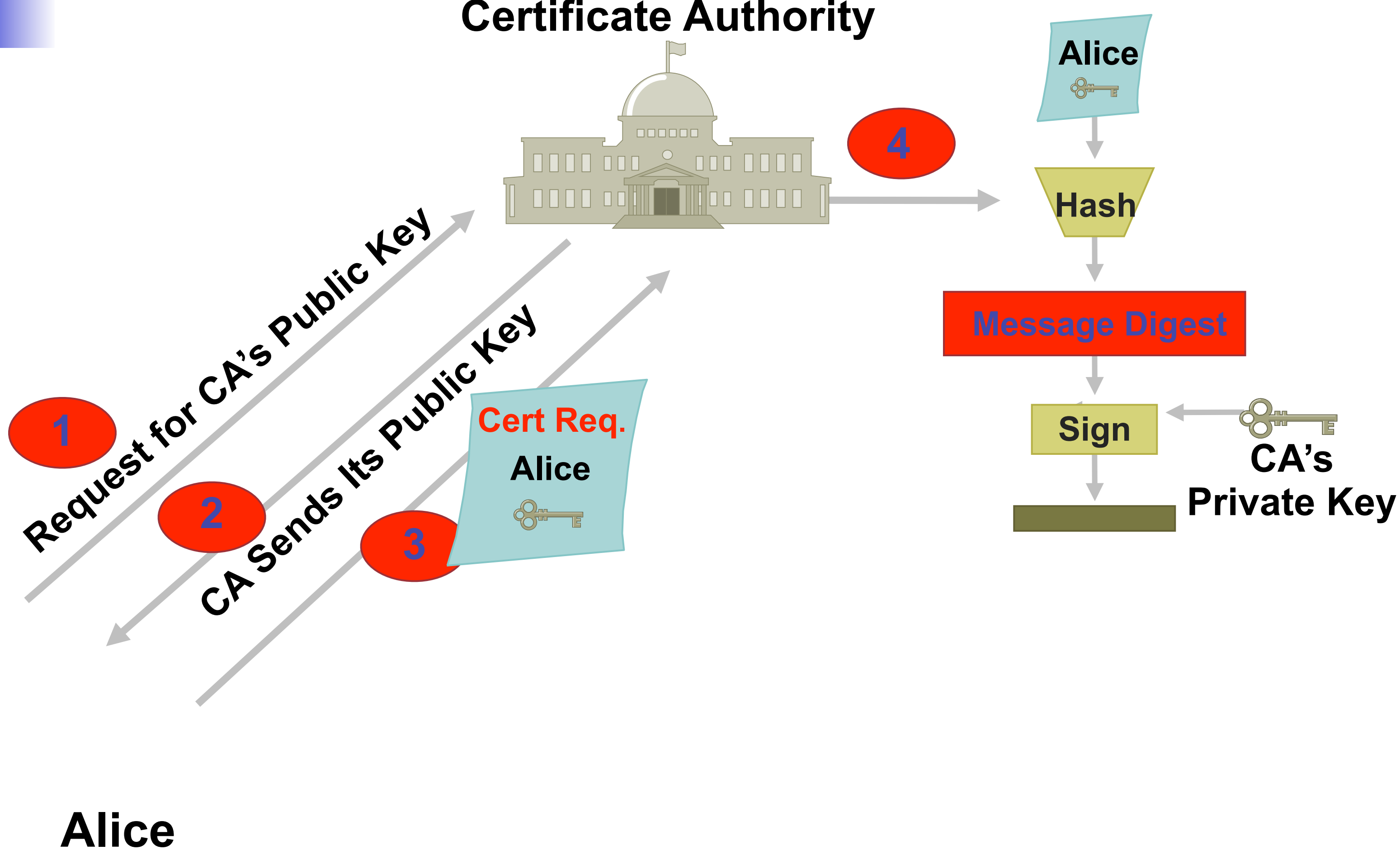


Alice

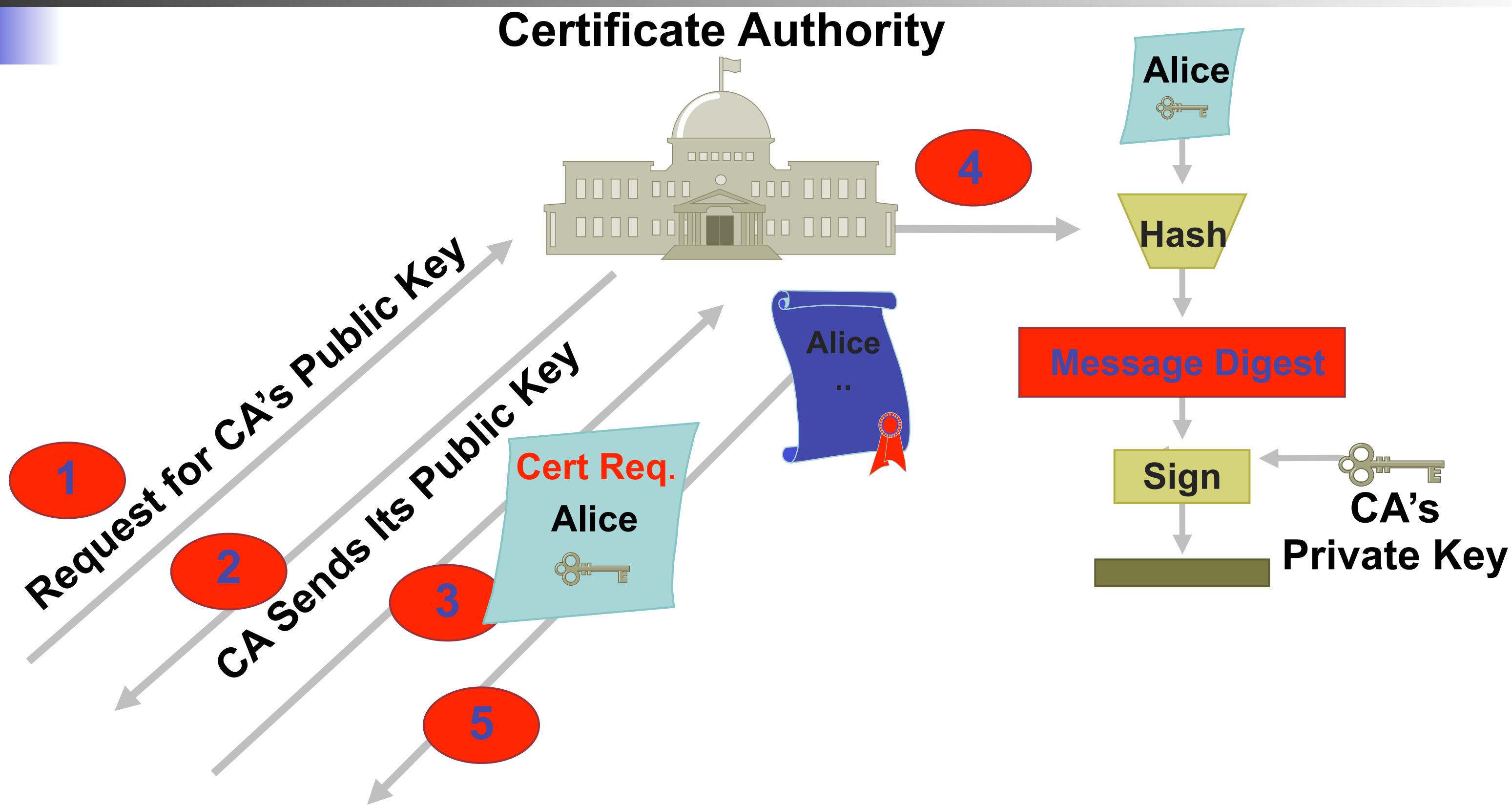
Digital Certification



Digital Certification

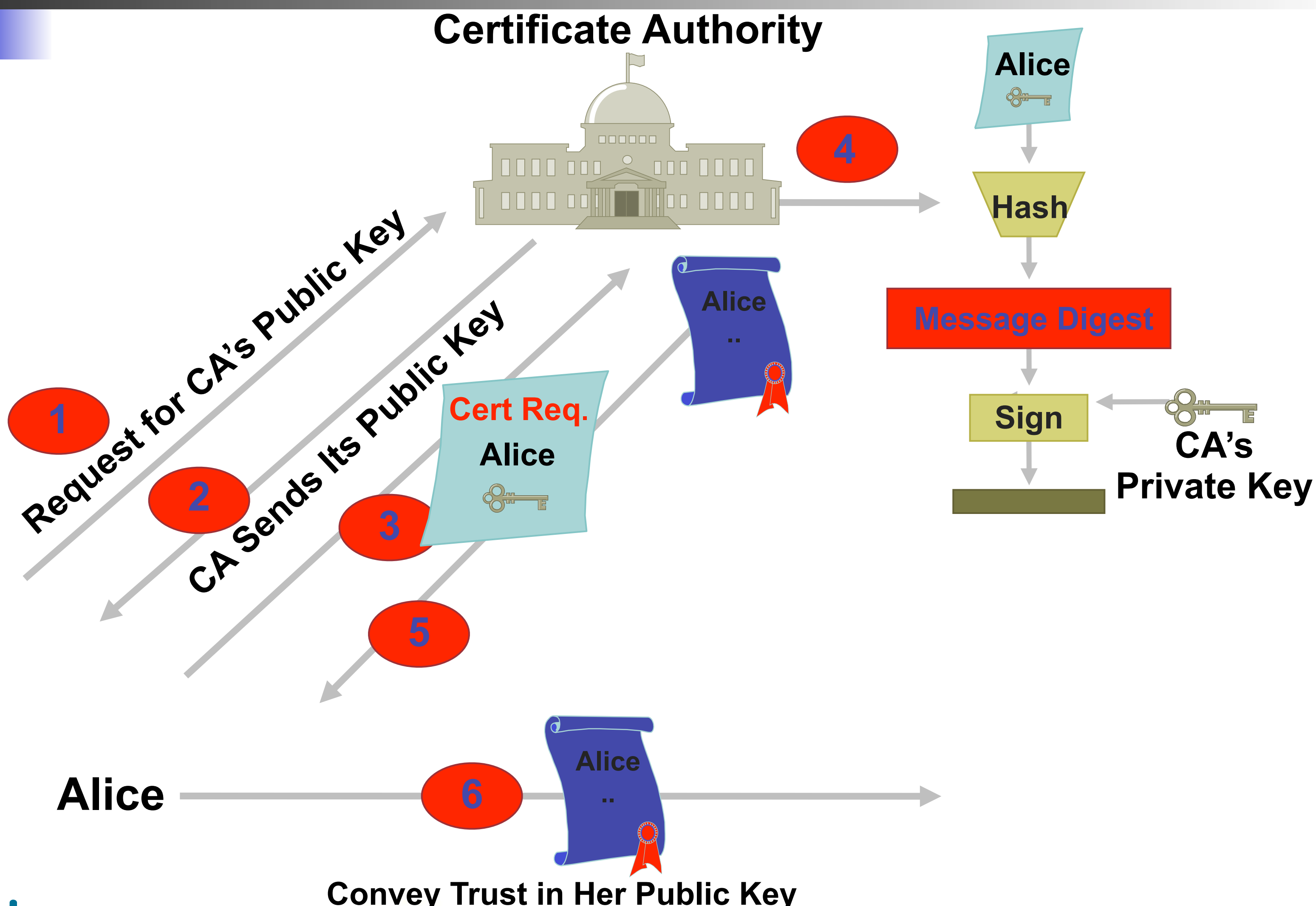


Digital Certification

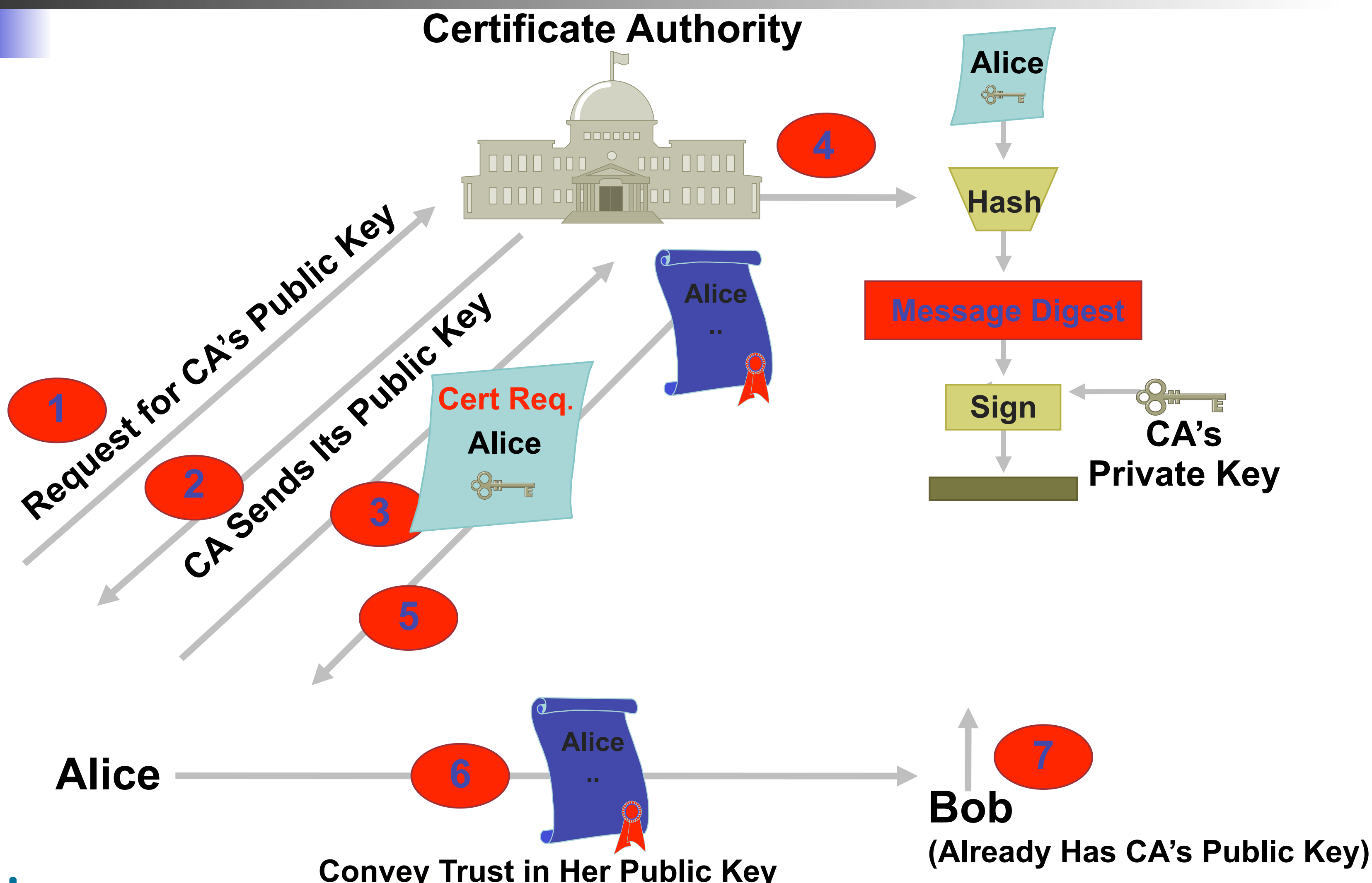


Alice

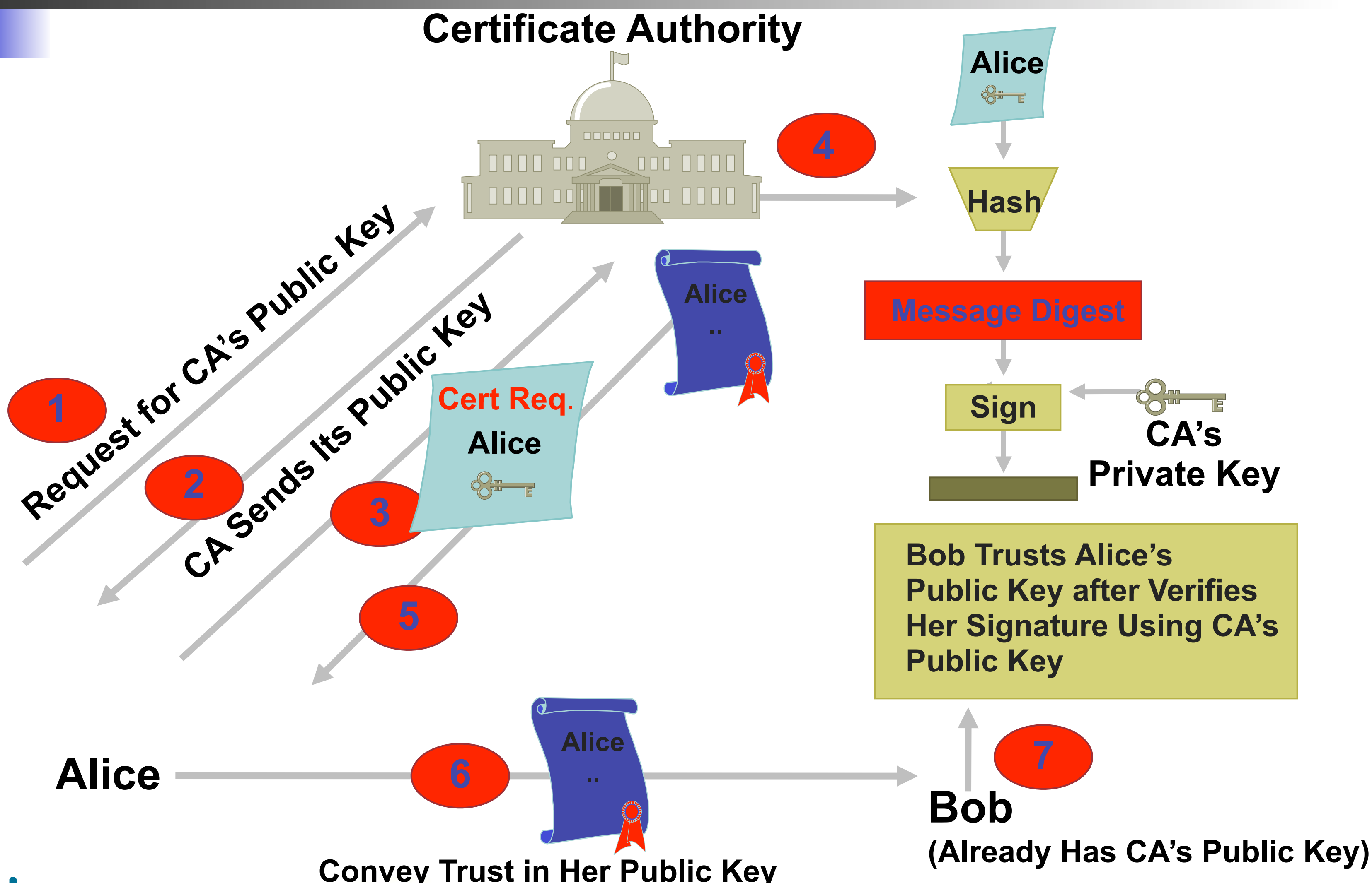
Digital Certification



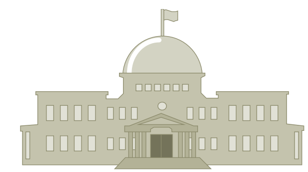
Digital Certification



Digital Certification



Certificate Authority



- The trust basis of a PKI system
- Verify user identity, issues certificates by binding user's identity to a public key with a digital certificate
- Revokes certificates and publish Certificate Revocation List (CRL)
- In-house implementation or outsourcing

Registration Authority (RA)

- An RA provides interface between the user and CA
  - For example, a CGI script
- Publishes CRL

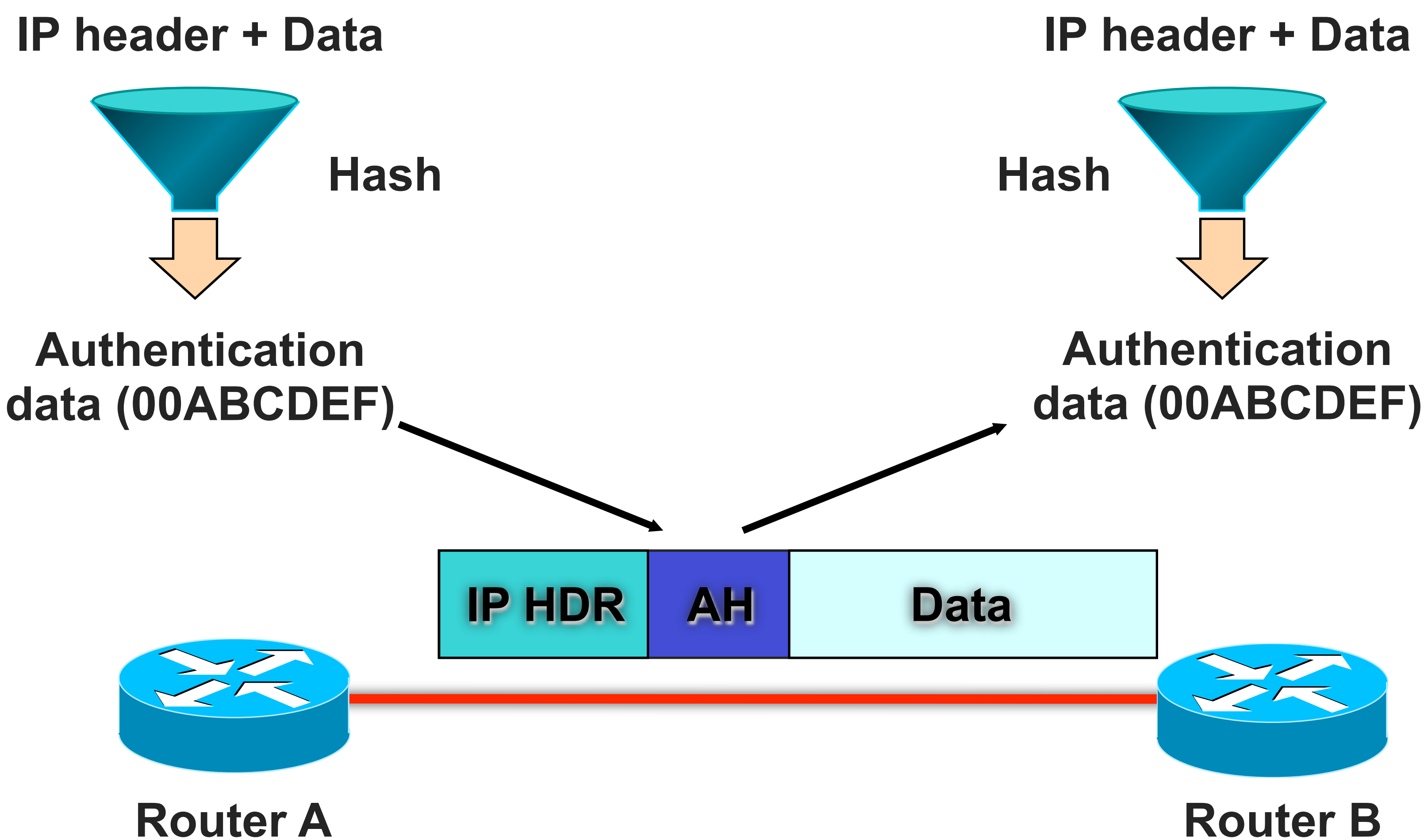
Certificate Revocation List (CRL)

- Certificates can be revoked by CA
  - Key compromise
  - Cessation of operation
- CRL is a list of the serial numbers of revoked certificates
- Makes PKI scalable
- CRL is published by CA or RA
- CRL also has a lifetime and is updated frequently by CA

Authentication Header

- Ensures data integrity
- Provides origin authentication—ensures packets definitely came from peer router
- Uses keyed-hash mechanism
- Does NOT provide confidentiality (no encryption)
- Provides optional replay protection

AH Authentication and Integrity

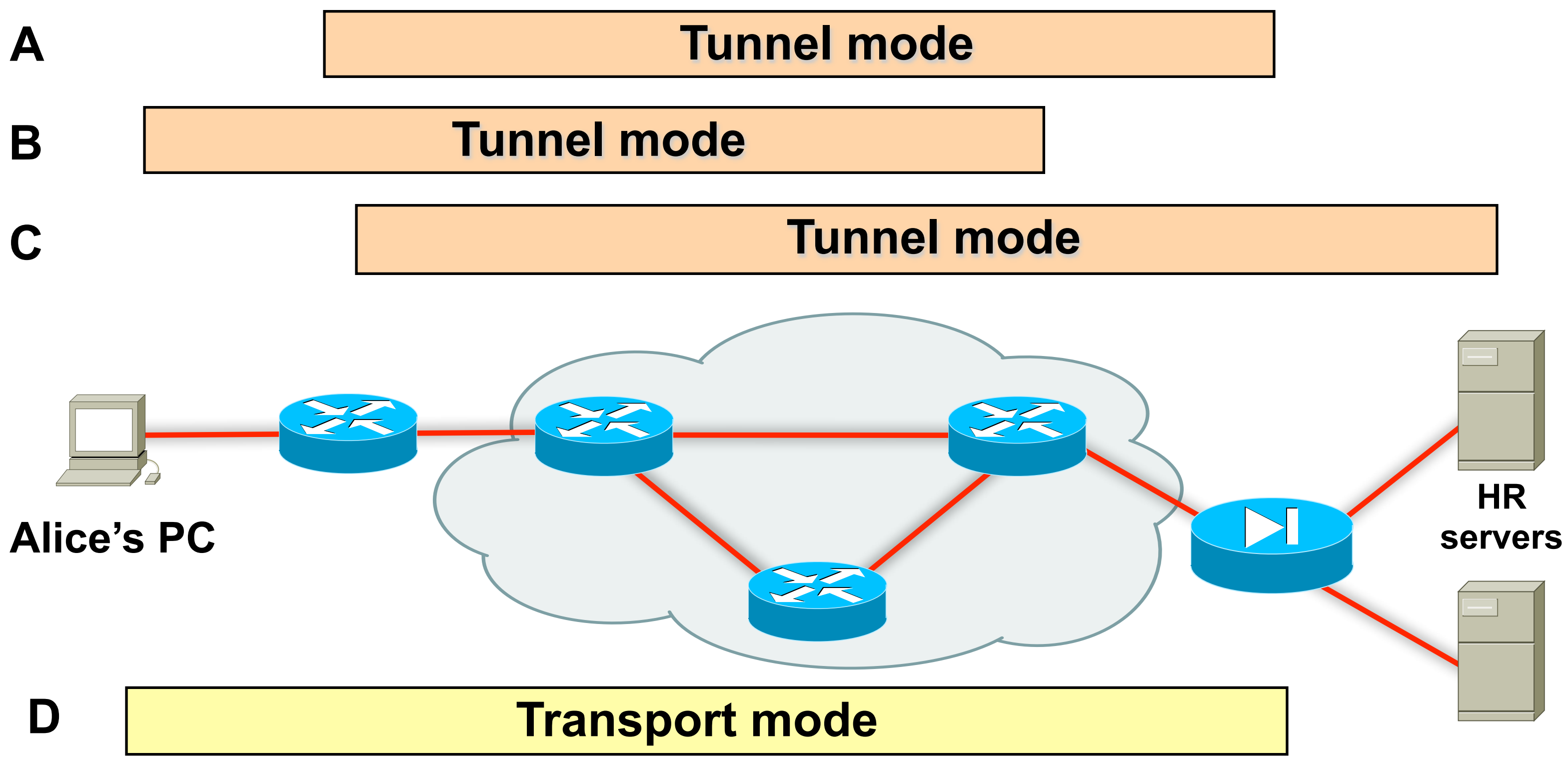


Encapsulating Security Payload

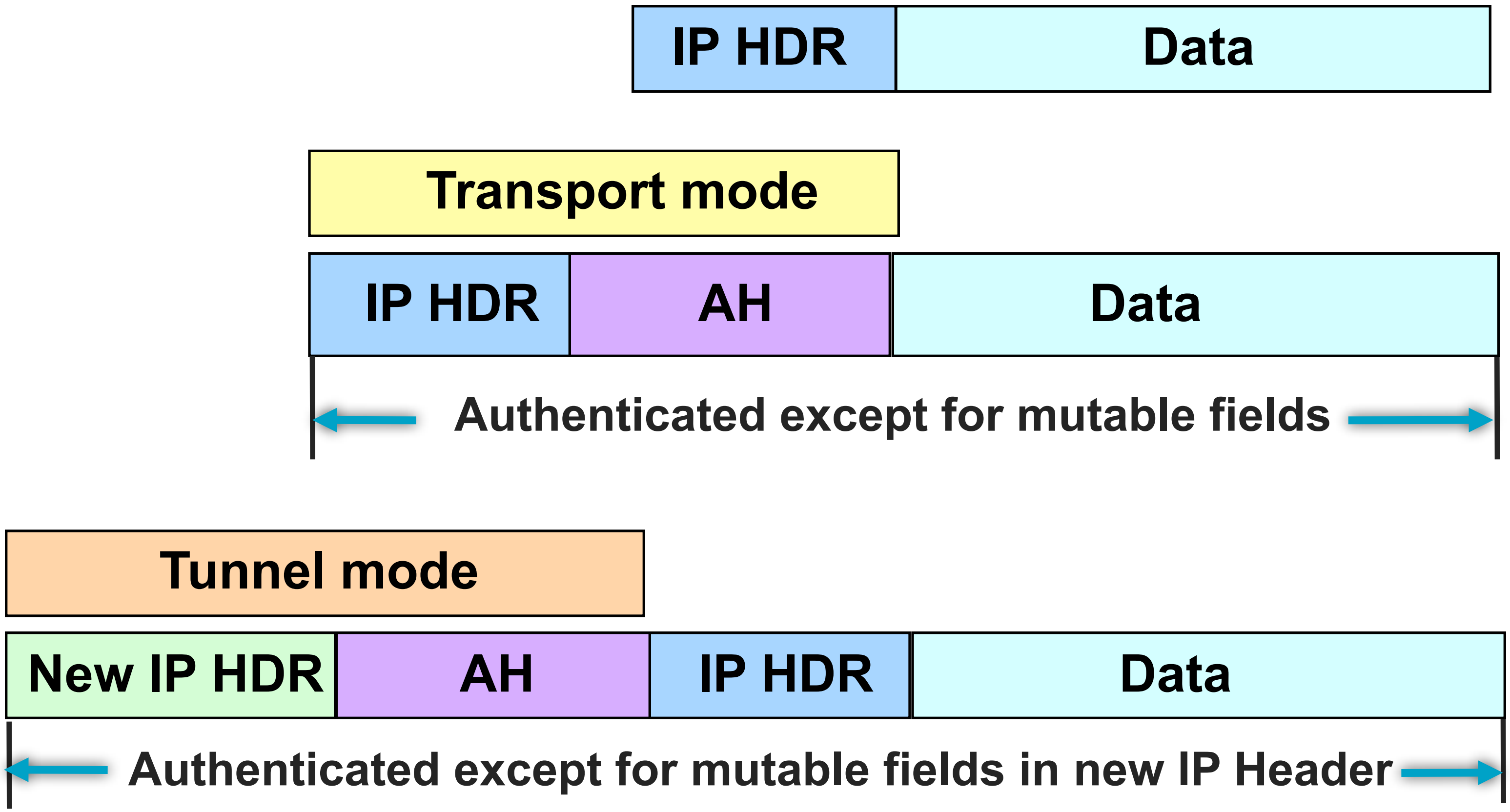
- Data confidentiality (encryption)
- Limited traffic flow confidentiality
- Data integrity
- Optional data origin authentication
- Anti-replay protection
- Does not protect IP header

Tunnel and Transport Modes

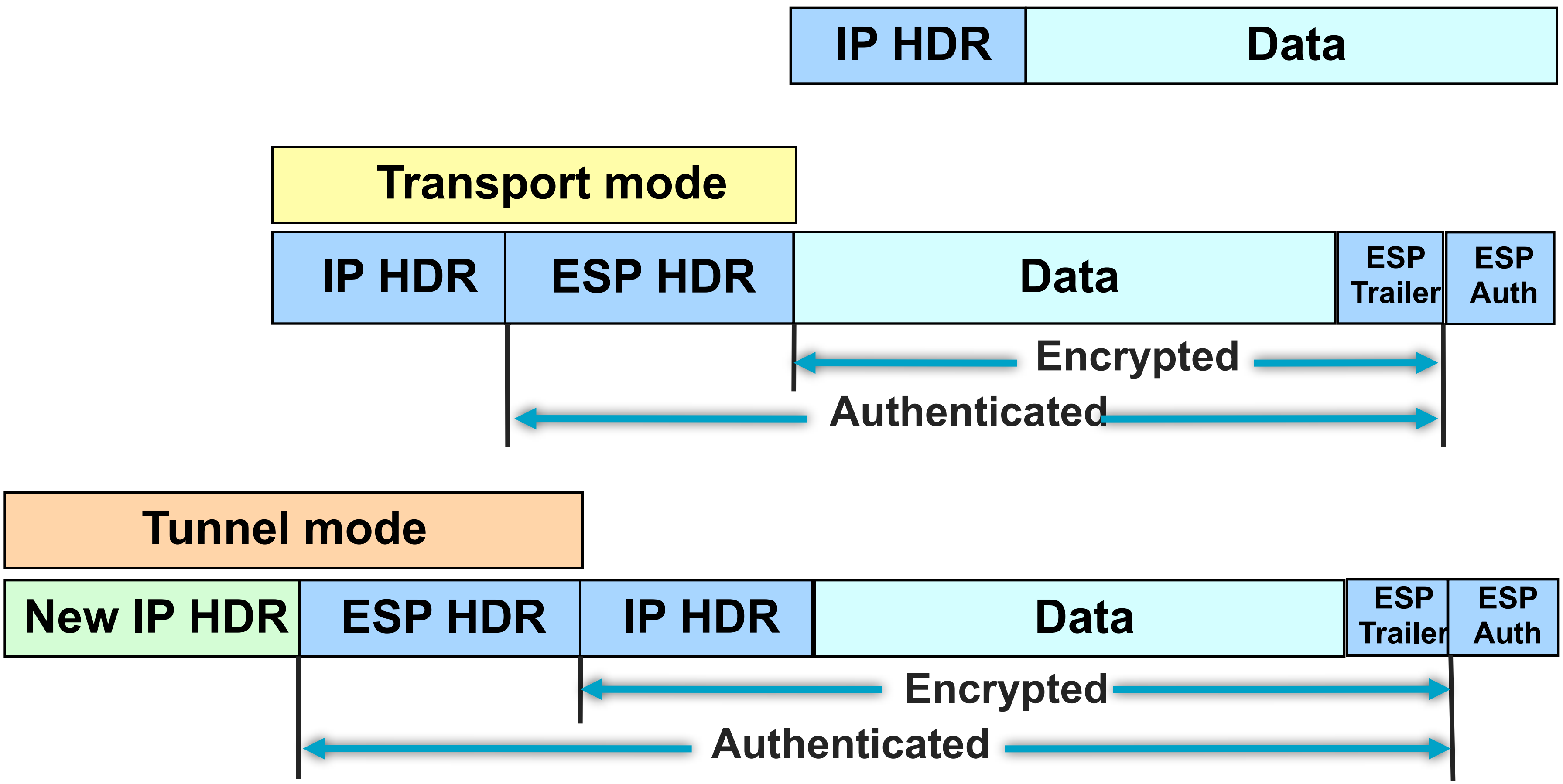
- Transport mode for end-to-end session
- Tunnel mode for everything else



AH Tunnel vs. Transport Mode

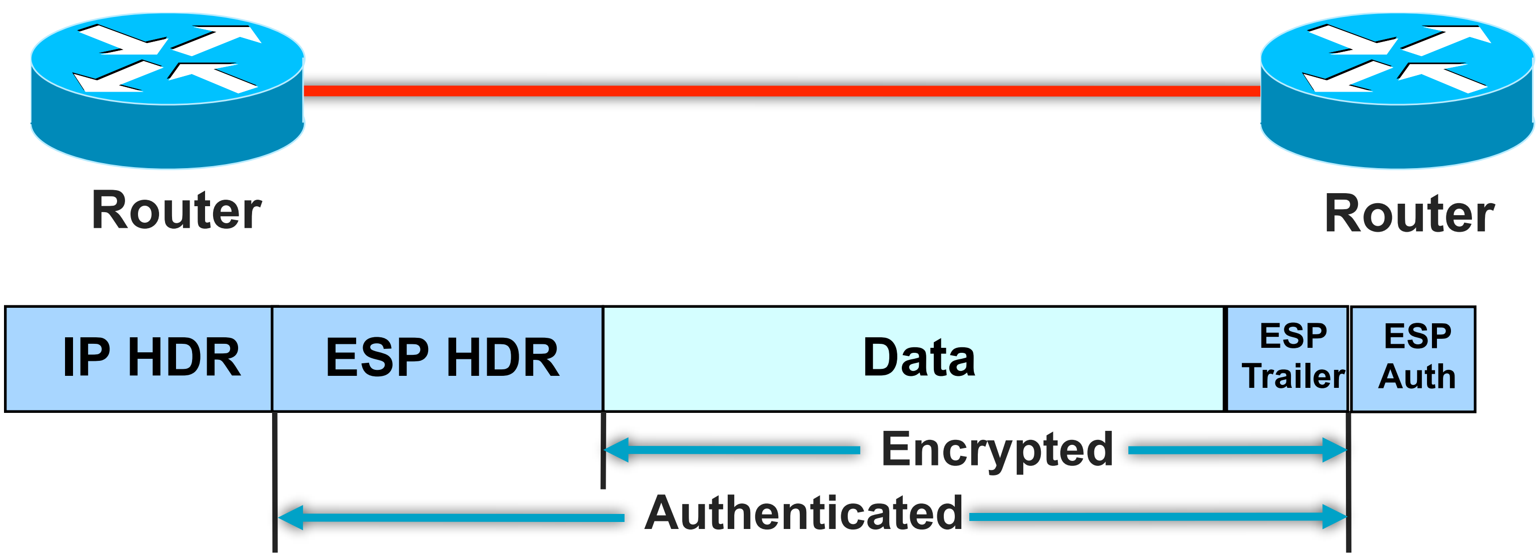


ESP Tunnel vs. Transport Mode



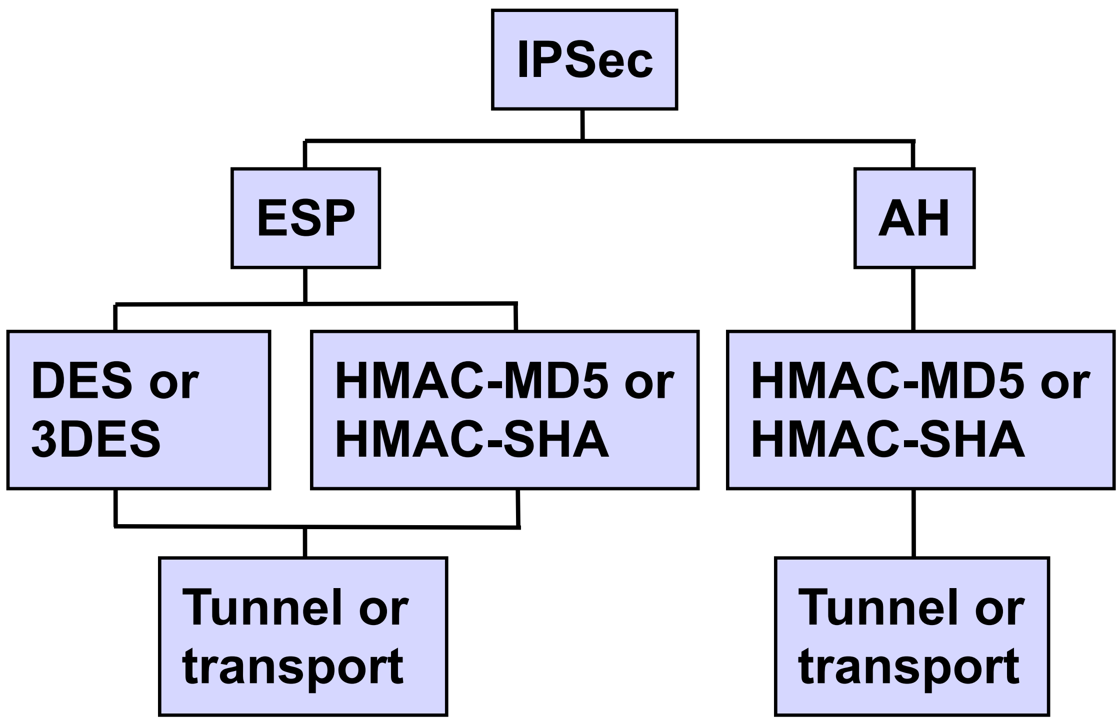
ESP Encryption with a Keyed-HMAC

- Provides ESP confidentiality with encryption
- Provides integrity with a keyed HMAC



IPSec Transforms

An IPSec transform specifies either an AH or an ESP protocol and its corresponding algorithms and mode.



Transform Sets

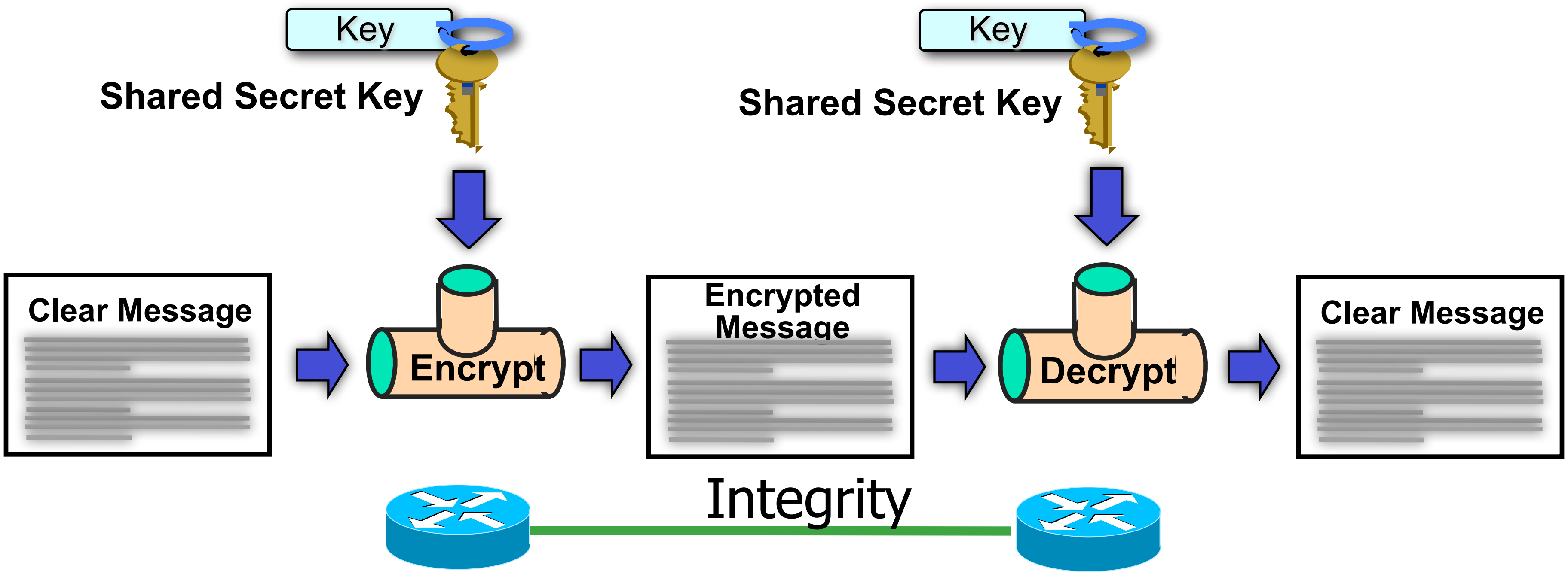
```
Transform1 + Transform2 + Transform3
esp-des
ah-md5-hmac
esp-md5-hmac + esp-des
esp-sha-hmac + esp-3des
ah-sha-hmac  + esp-3des + esp-sha-hmac
```

- **A transform set is a combination of IPSec transforms that enact a security policy for traffic**
- **Up to three transforms can be in a set**
- **Sets are limited to up to one AH and up to two ESP transforms**

Data Encryption Standard (DES)

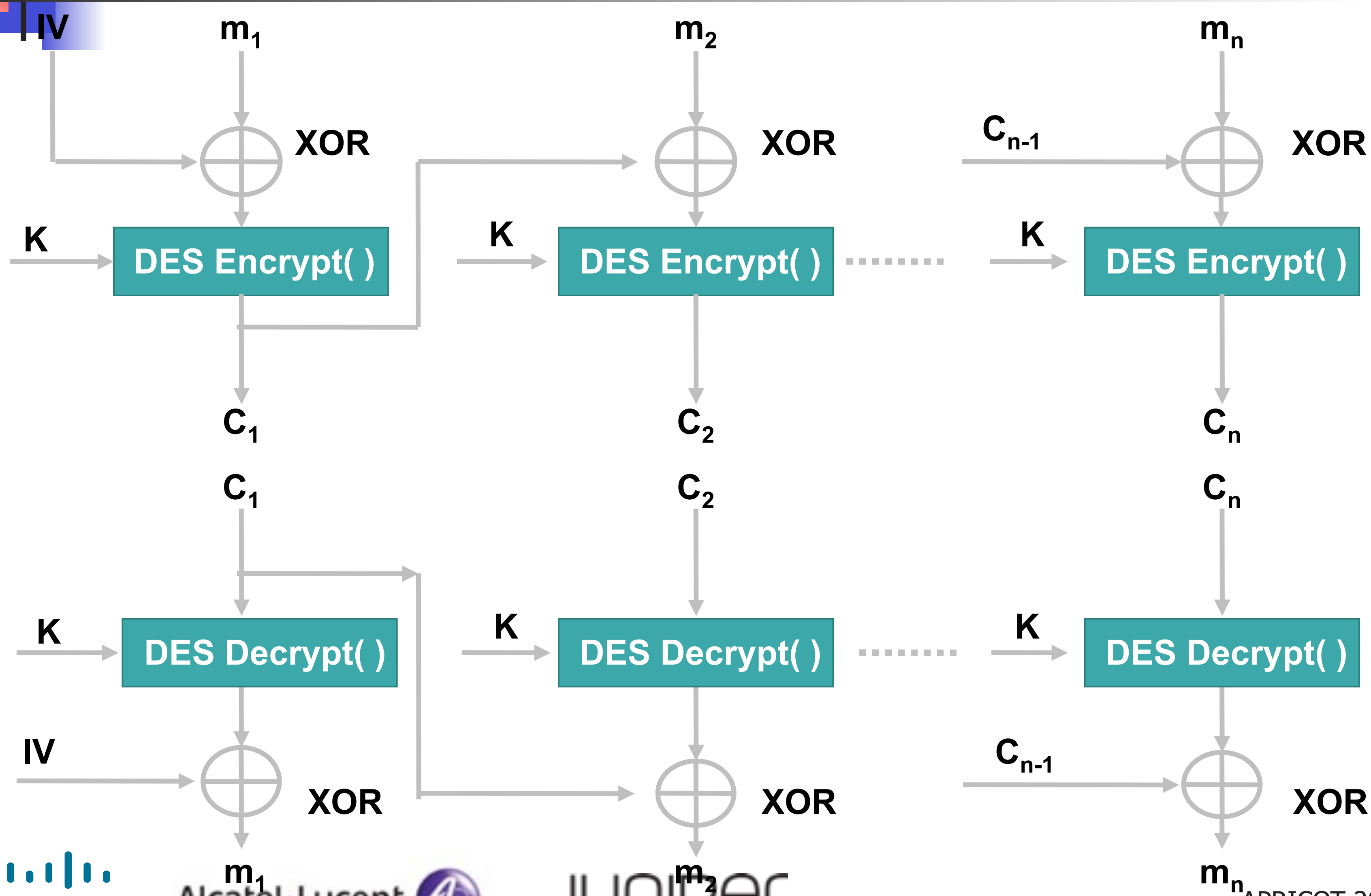
- Symmetric key encryption algorithm
- Block cipher: works on 64-bit data block, use 56-bit key (last bit of each byte used for parity)
- Mode of operation: how to apply DES to encrypt blocks of data
  - Electronic Code Book (ECB)
  - Cipher Block Chaining (CBC)
  - K-bit Cipher FeedBack (CFB)
  - K-bit Output FeedBack (OFB)

DES Encryption

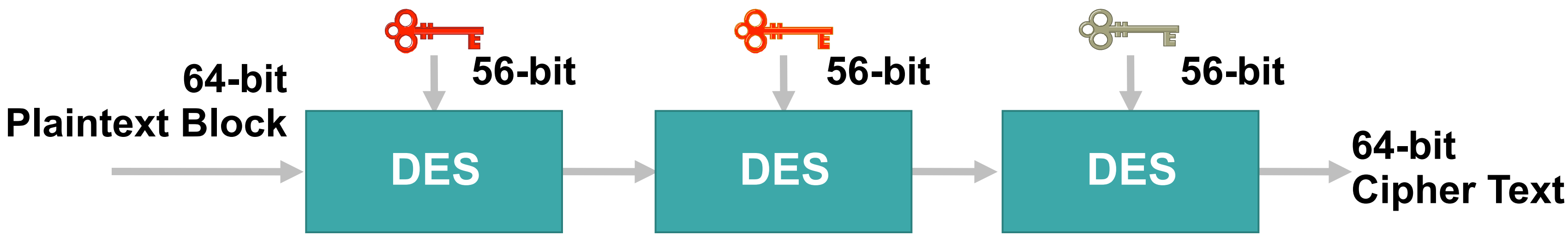


- Encryption turns cleartext into ciphertext.
- Decryption restores cleartext from ciphertext.
- Keys enable encryption and decryption.

DES CBC Mode


IV
m1
m2
mn
XOR
XOR
Cn-1
XOR
K
DES Encrypt( )
K
DES Encrypt( )
K
DES Encrypt( )
C1
C2
Cn
C1
C2
Cn
K
DES Decrypt( )
K
DES Decrypt( )
K
DES Decrypt( )
IV
XOR
XOR
Cn-1
XOR
m1
m2
mn

Triple-DES



- 168-bit total key length
- Mode of operation decides how to process DES three times
- Normally: encrypt, decrypt, encrypt
- More secure than DES but slower
- So is 3DES optimally the fastest, the easiest to implement and the securest algorithm out there?

AES Encryption

- Published in November 2001
- Rijndael algorithm developed by Dr. Joan Daemen and Dr. Vincent Rijmen
- Symmetric Block Cipher
- 128 bit blocks
- 3 key lengths: 128, 192, and 256 bits
- Symmetric and parallel
- Low memory requirement

AES Key Length

- **Key Length (in bits) Number of Combinations**
  - 40 240 = 1,099,511,627,776
  - 56 256 = 7.2 x 1016
  - 64 264 = 1.8 x 1019
  - 112 2112 = 5.2 x 1033
  - 128 2128 = 3.4 x 1038
  - 192 2192 = 6.2 x 1057
  - 256 2256 = 1.1 x 1077

Diffie-Hellman Key Agreement

Peer A  Peer B

Peer A

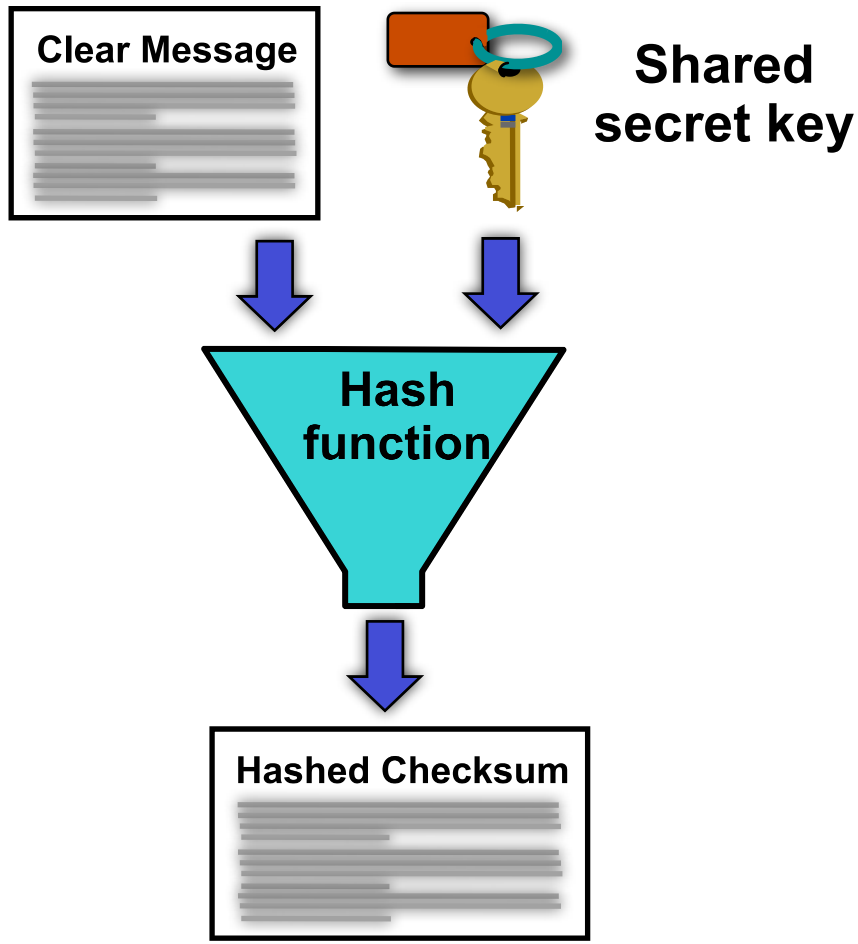
1. Generate large integer p.
   Send p to Peer B.
   Receive q.
   Generate g.
2. Generate private key X_A
3. Generate public key
   $Y_A = g \wedge X_A \bmod p$
4. Send public key Y_A
5. Generate shared secret number $ZZ = Y_B \wedge X_A \bmod p$
6. Generate shared secret key from ZZ (56-bit for DES, 168-bit for 3DES)

Peer B

1. Generate large integer q.
   Send q to Peer A.
   Receive p.
   Generate g.
2. Generate private key X_B
3. Generate public key
   $Y_B = g \wedge X_B \bmod p$
4. Send public key Y_B
5. Generate shared secret number $ZZ = Y_A \wedge X_B \bmod p$
6. Generate shared secret key from ZZ (56-bit for DES, 168-bit for 3DES)

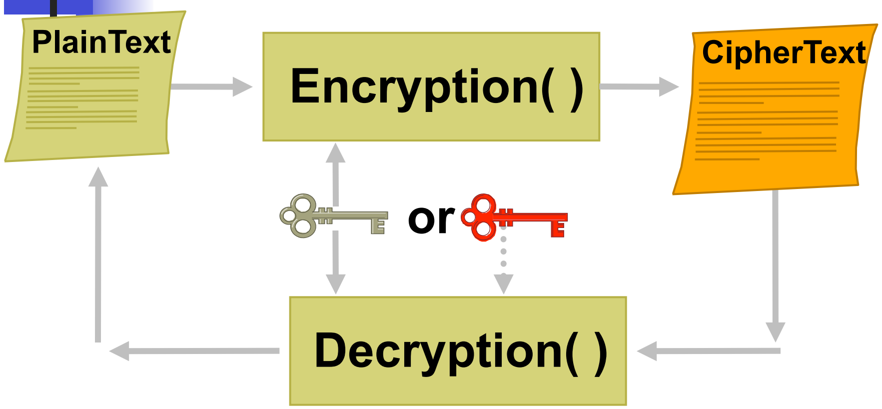
Hashed Message Authentication Codes (HMAC)

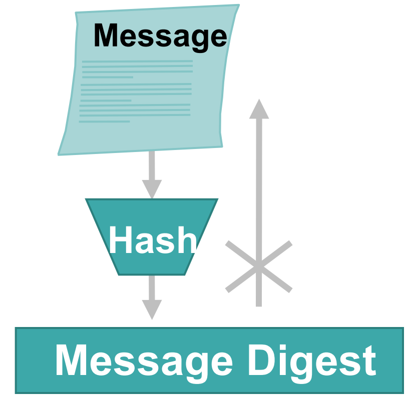
- Variable-length input message



Fixed-length authenticator value

Encryption vs. Hashing





- Encryption keeps communications private
- Encryption and decryption can use same or different keys
- Achieved by various algorithms, e.g. DES, CAST
- Need key management

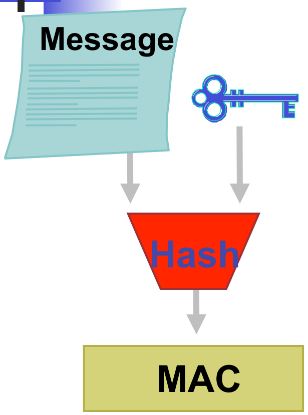
- Hash transforms message into fixed-size string
- One-way hash function
- Strongly collision-free hash
- Message digest can be viewed as "digital fingerprint"
- Used for message integrity check and digital certificates
- Hash is generally faster than encryption

Message Authentication and Integrity Check Using Hash

Sender **Receiver**

Secret Key Only Known by Sender and Receiver

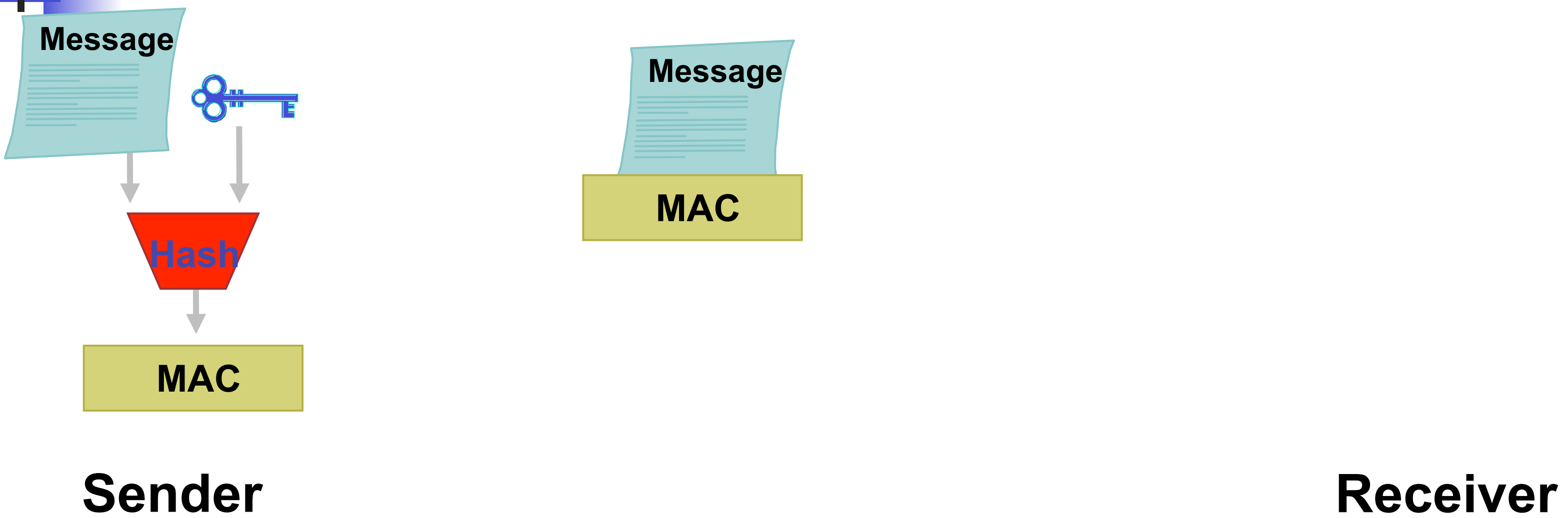
Message Authentication and Integrity Check Using Hash



Sender **Receiver**

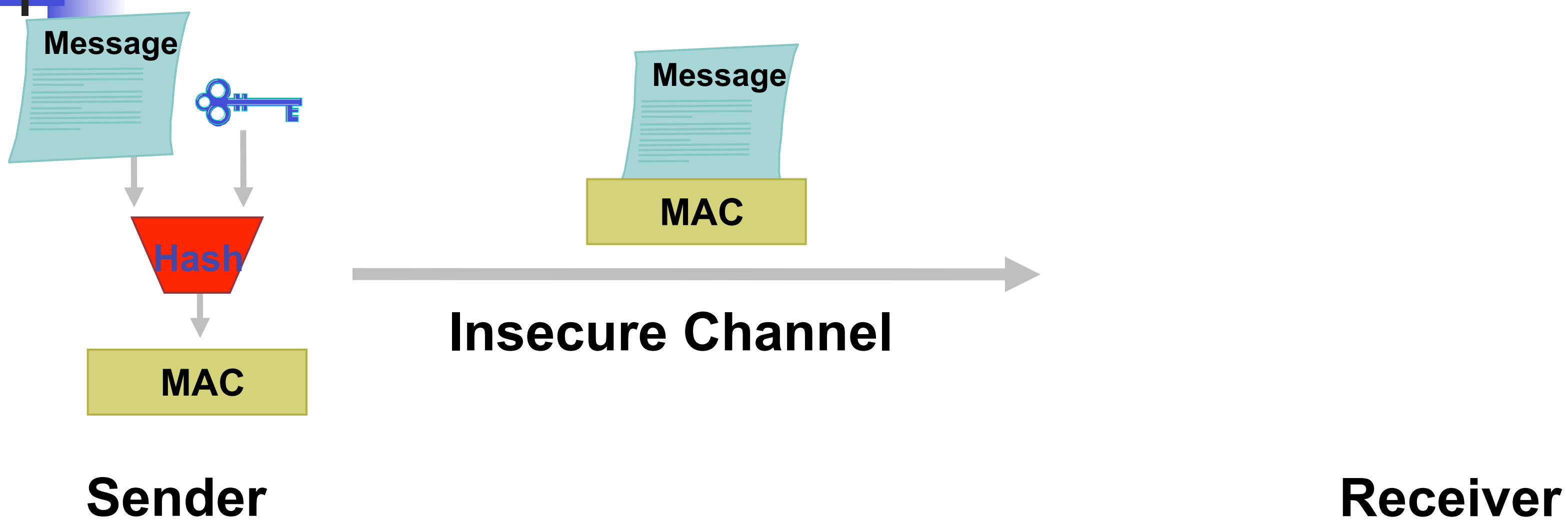
Secret Key Only Known by Sender and Receiver

Message Authentication and Integrity Check Using Hash



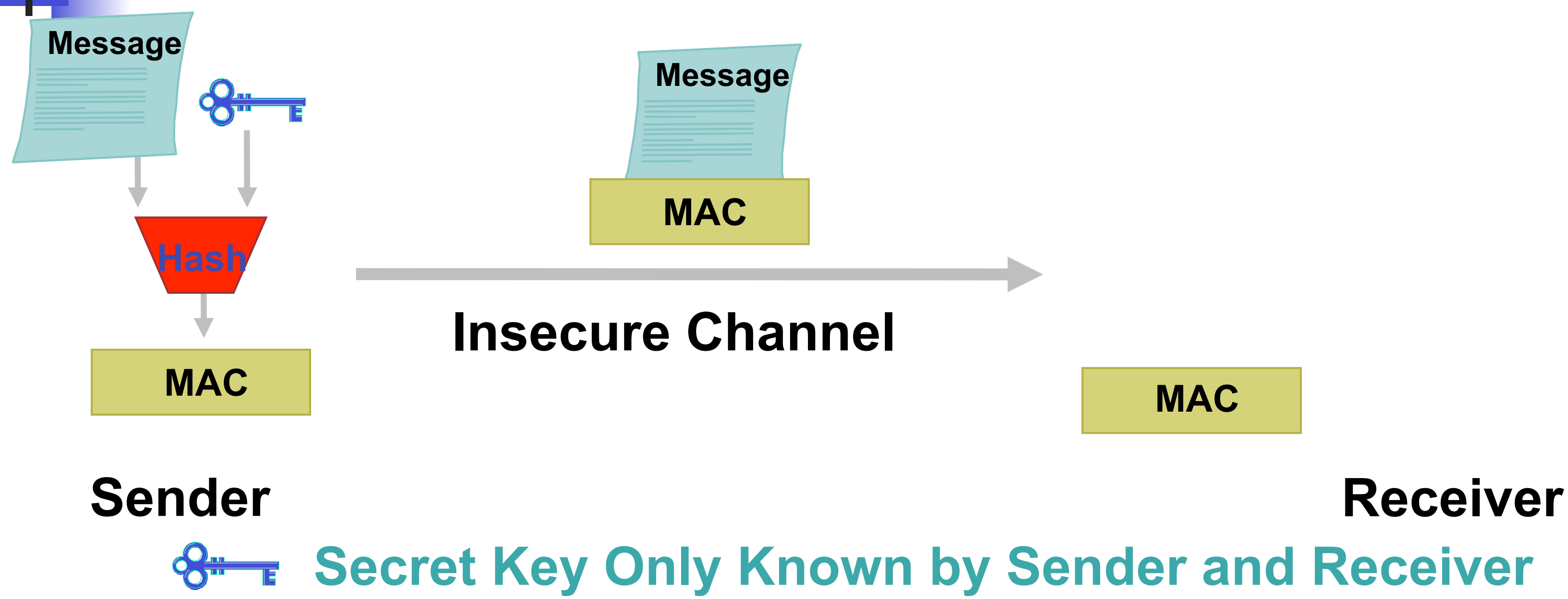
Secret Key Only Known by Sender and Receiver

Message Authentication and Integrity Check Using Hash

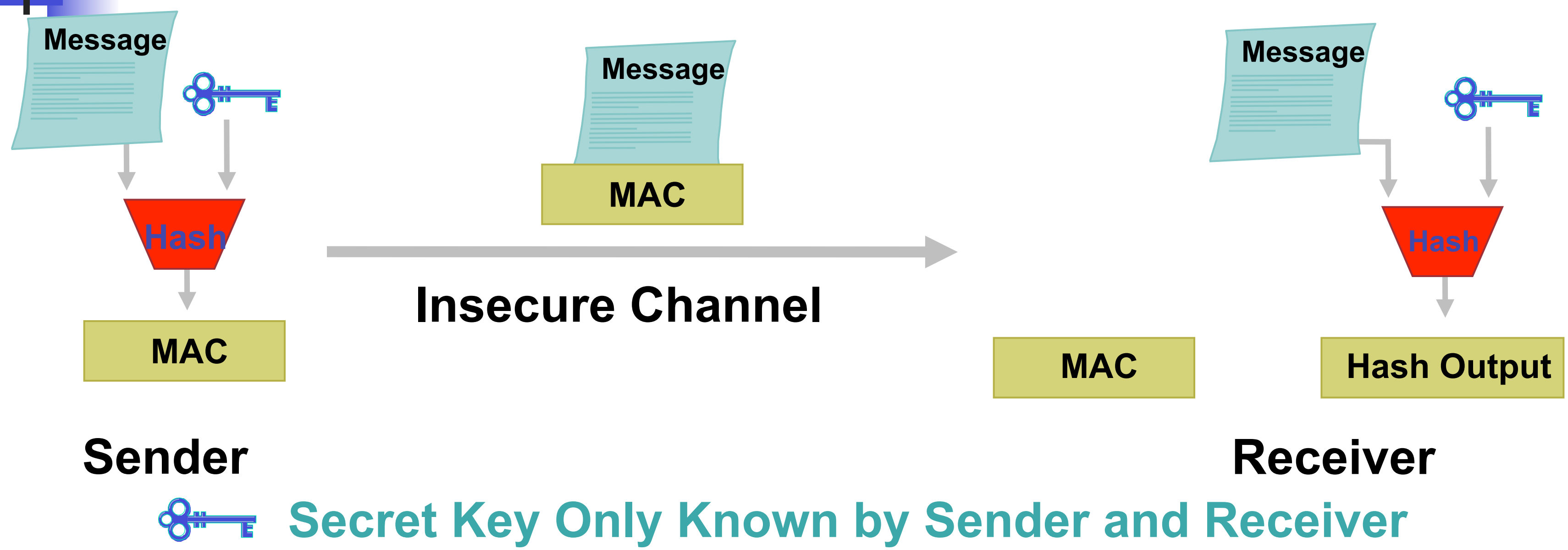


Secret Key Only Known by Sender and Receiver

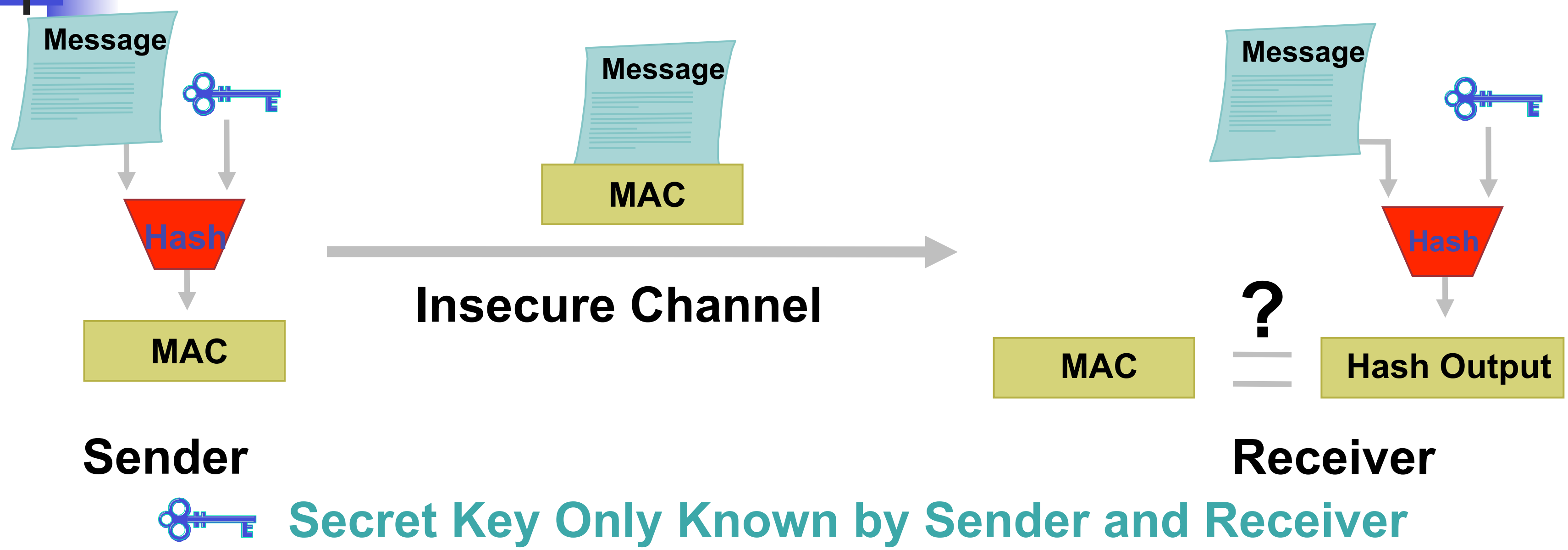
Message Authentication and Integrity Check Using Hash



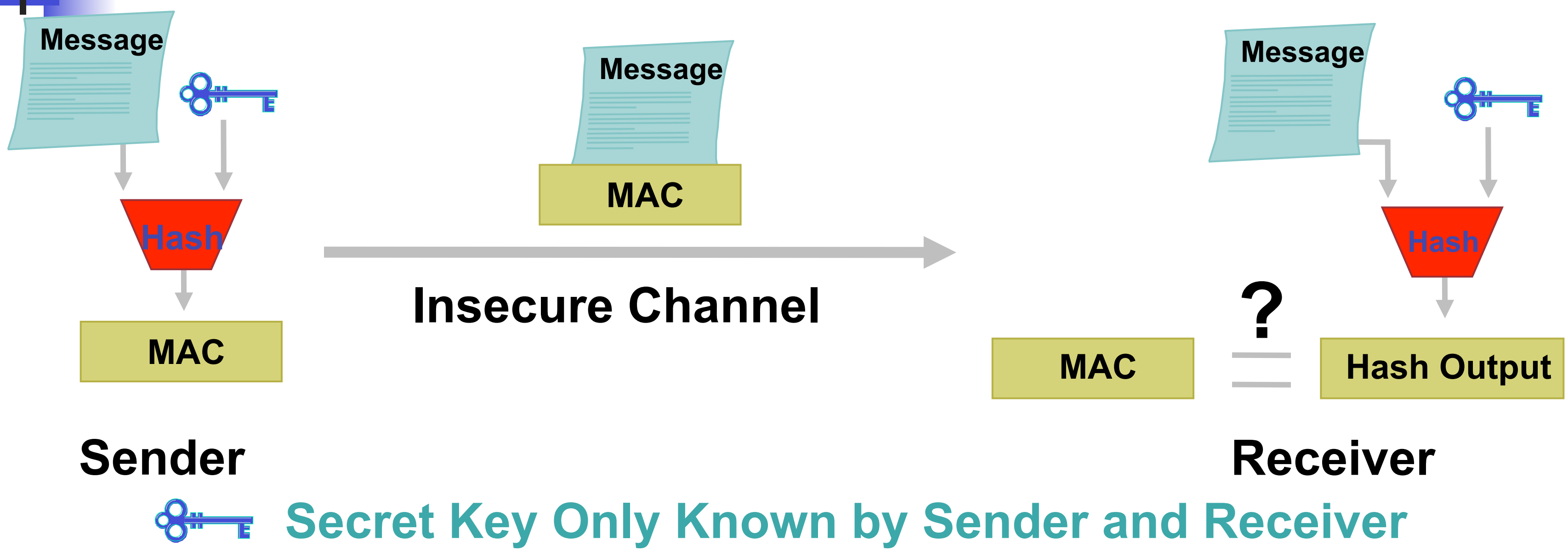
Message Authentication and Integrity Check Using Hash



Message Authentication and Integrity Check Using Hash

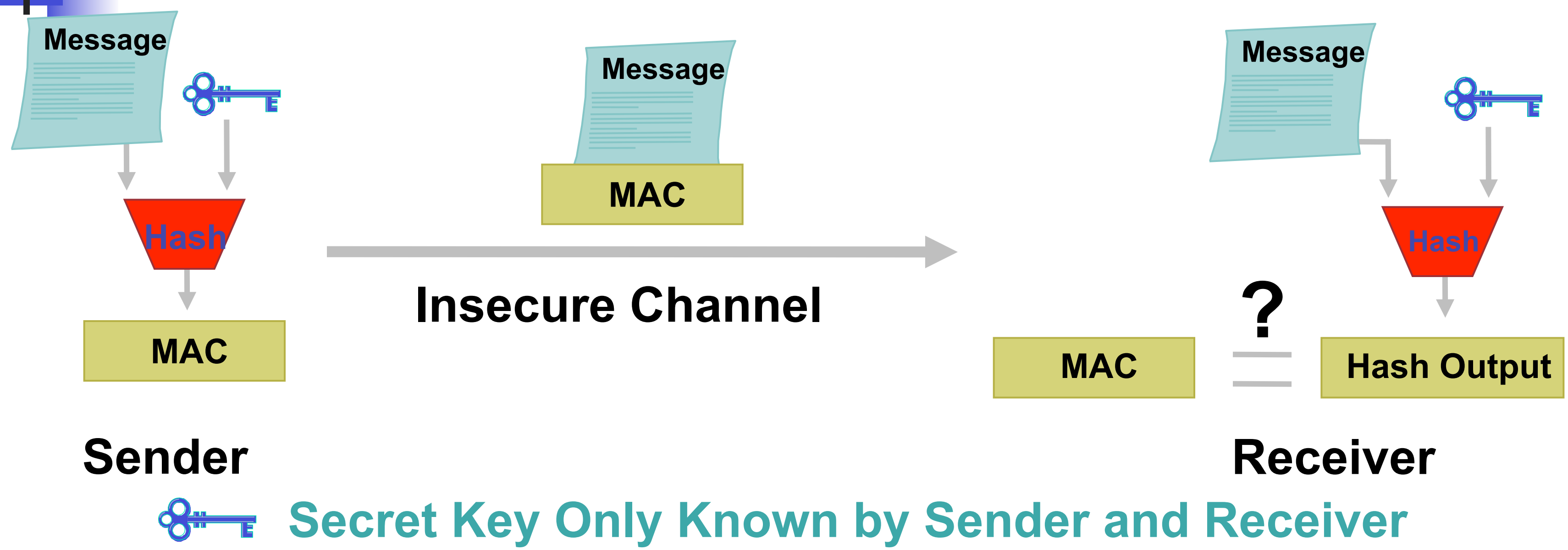


Message Authentication and Integrity Check Using Hash



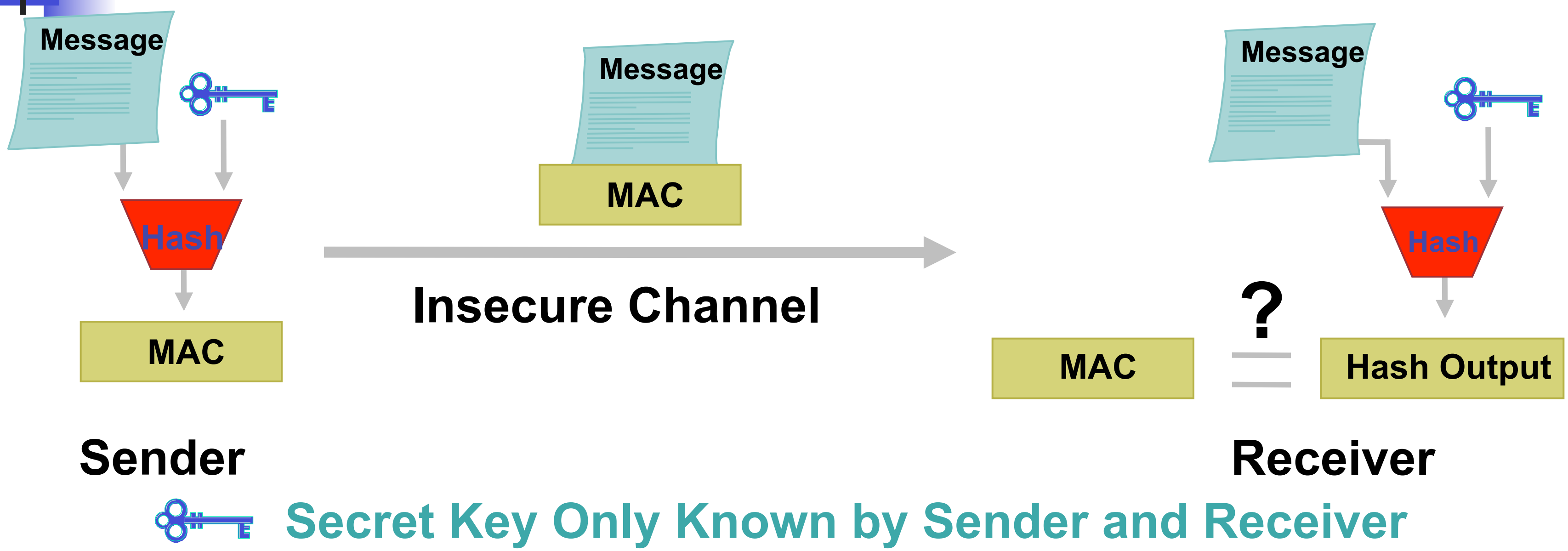
- MAC (Message Authentication Code): cryptographic checksum generated by passing data thru a message authentication algorithm

Message Authentication and Integrity Check Using Hash



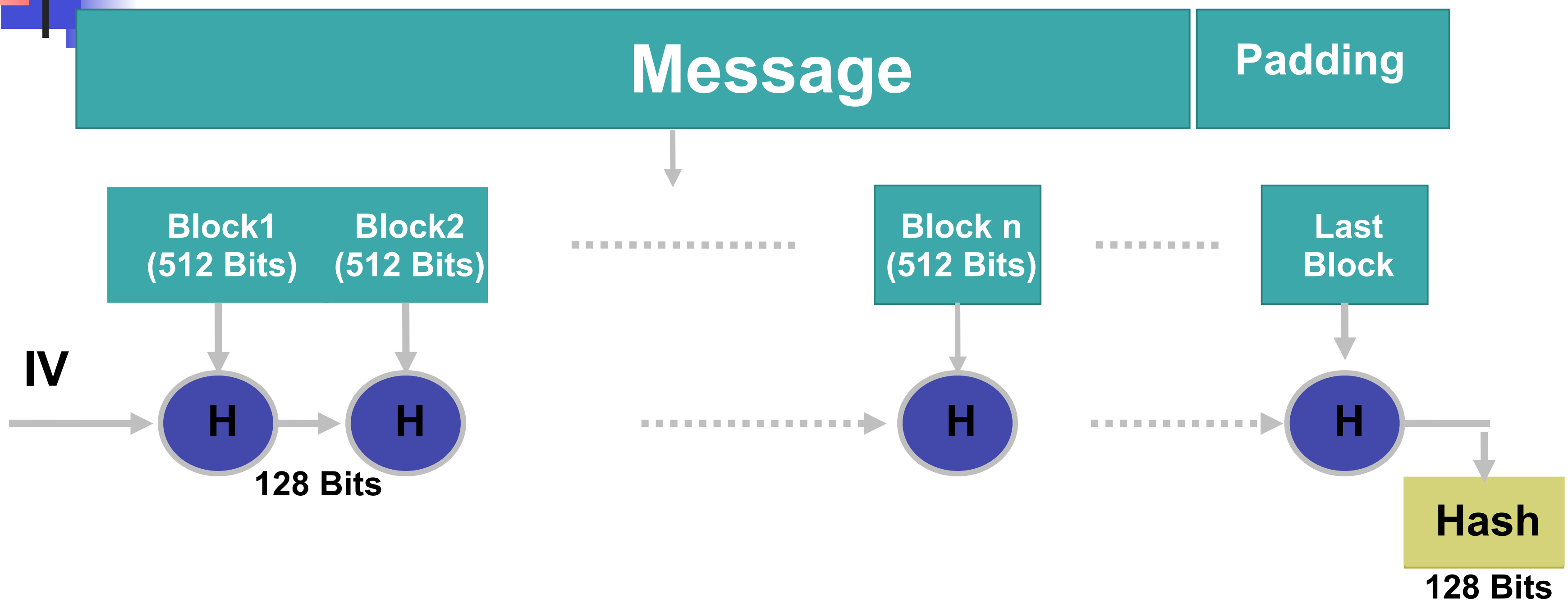
- MAC (Message Authentication Code): cryptographic checksum generated by passing data thru a message authentication algorithm
- MAC is often used for message authentication and integrity check

Message Authentication and Integrity Check Using Hash



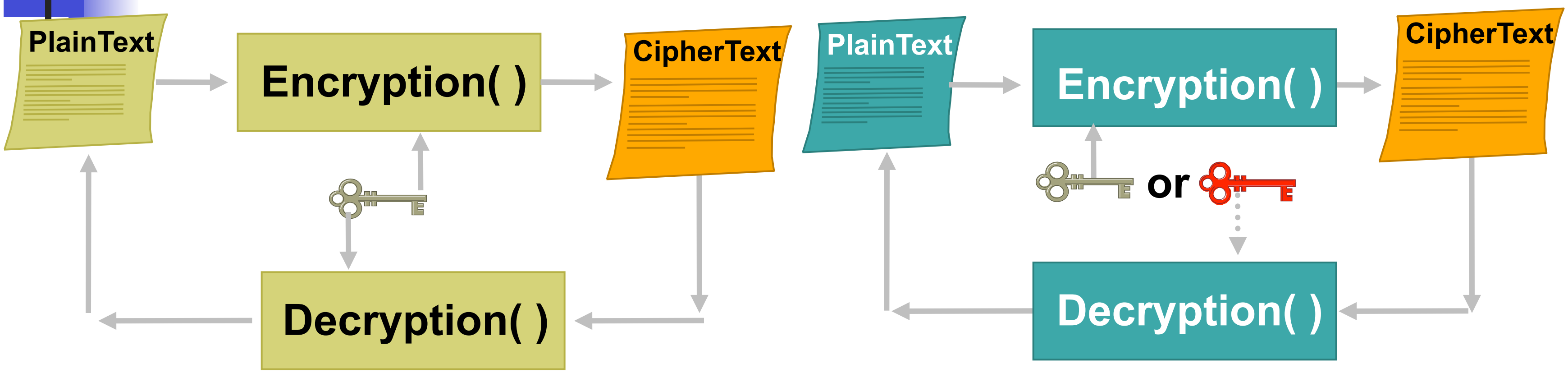
- MAC (Message Authentication Code): cryptographic checksum generated by passing data thru a message authentication algorithm
- MAC is often used for message authentication and integrity check
- HMAC—keyed hashed-based MAC

Commonly Used Hash Functions (MD5 and SHA)



- Both MD5 and SHA are derived based on MD4
- MD5 provides 128-bit output, SHA provide 160-bit output; (only first 96 bits used in IPSec)
- Both of MD5 and SHA are considered one-way strongly collision-free hash functions
- SHA is computationally slower than MD5, but more secure

Symmetric vs. Asymmetric Encryption Algorithms



- Secret-key cryptography
- Encryption and decryption use the same key
- Typically used to encrypt the content of a message
- Examples: DES

- Public-key cryptography
- Encryption and decryption use different keys
- Typically used in digital certification and key management
- Examples: Diffie-Hellman, RSA

Five Steps of IPSec

1. Host A sends interesting traffic to Host B.

2. Router A and B negotiate an IKE phase one session.

IKE SA **IKE SA**

3. Router A and B negotiate an IKE phase two session.

IPSec SA **IPSec SA**

4. Information is exchanged via IPSec tunnel.

IPSec Tunnel

5. IPSec tunnel is terminated.

Five Steps of IPSec

1. Host A sends interesting traffic to Host B.

2. Router A and B negotiate an IKE phase one session.

IKE SA IKE SA

3. Router A and B negotiate an IKE phase two session.

IPSec SA IPSec SA

4. Information is exchanged via IPSec tunnel.

5. IPSec tunnel is terminated.

Five Steps of IPSec

1. Host A sends interesting traffic to Host B.

2. Router A and B negotiate an IKE phase one session.

IKE SA — IKE Phase 1 — IKE SA

3. Router A and B negotiate an IKE phase two session.

IPSec SA — IKE Phase 2 — IPSec SA

4. Information is exchanged via IPSec tunnel.

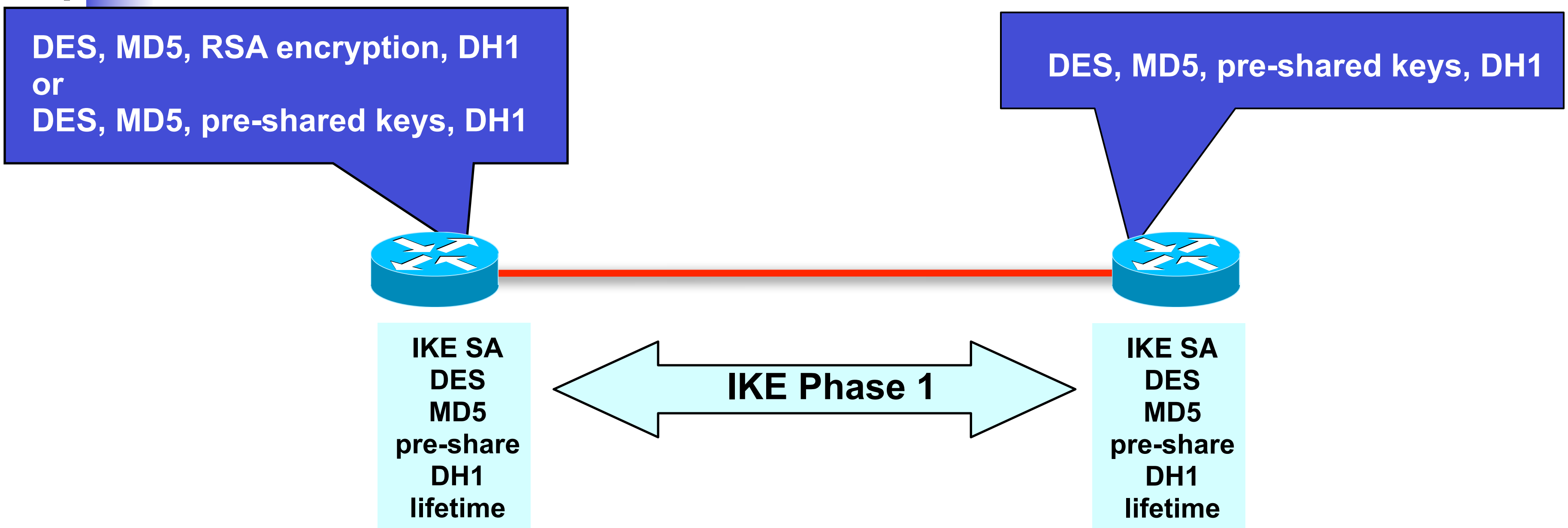
5. IPSec tunnel is terminated.

Step 1—Interesting Traffic

access-list 101 permit ip 10.0.1.0 0.0.0.255 10.0.2.0 0.0.0.255

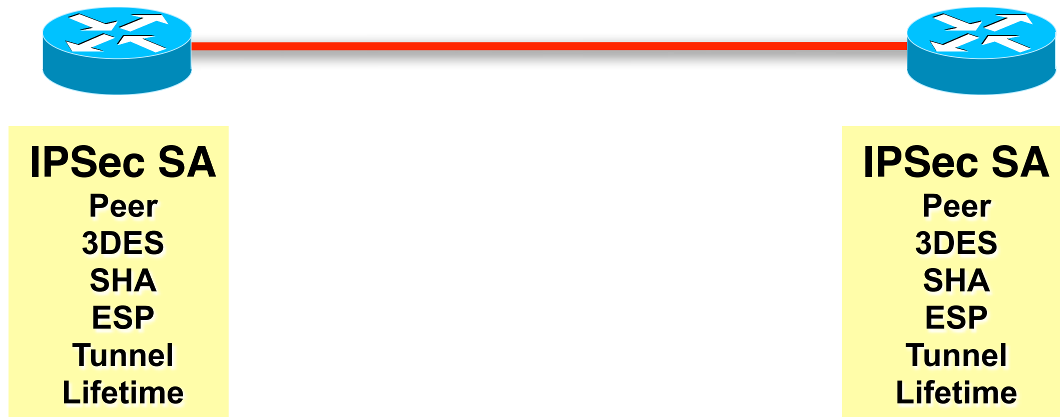
- Access lists determine traffic to encrypt
  - Permit—traffic must be encrypted
  - Deny—traffic sent unencrypted

Step 2—IKE Phase One



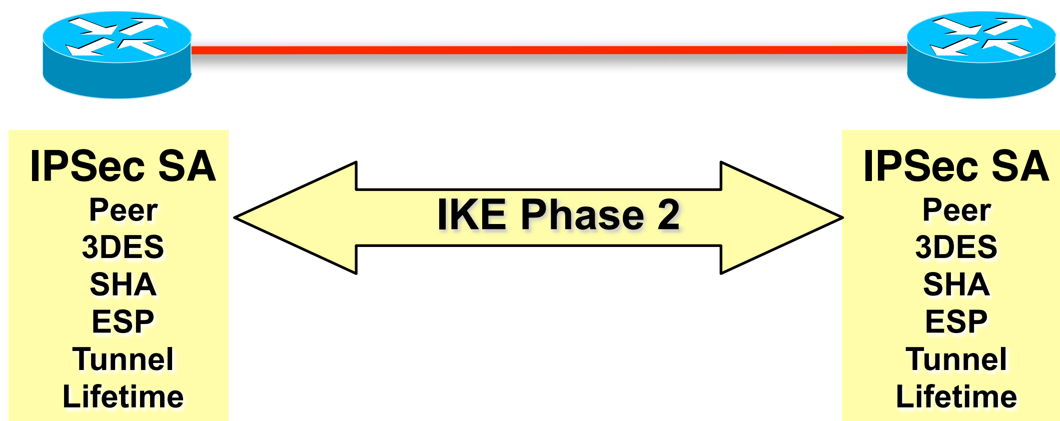
- Authenticates IPSec peers
- Negotiates matching policy to protect IKE exchange
- Exchanges keys via Diffie-Hellman
- Establishes IKE security association

Step 3—IKE Phase Two



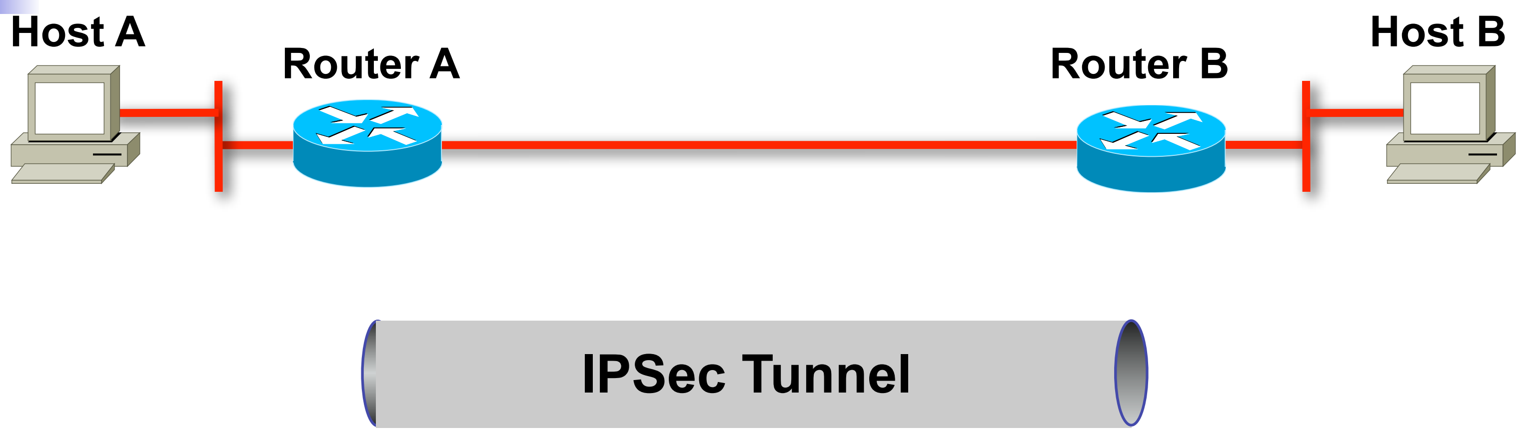
- Negotiates IPSec SA parameters protected by an existing IKE SA
- Establishes IPSec security associations
- Periodically renegotiates IPSec SAs to ensure security
- Optionally performs an additional Diffie-Hellman exchange
- If perfect forward secrecy is specified, a new Diffie-Hellman exchange is performed with each quick mode.

Step 3—IKE Phase Two



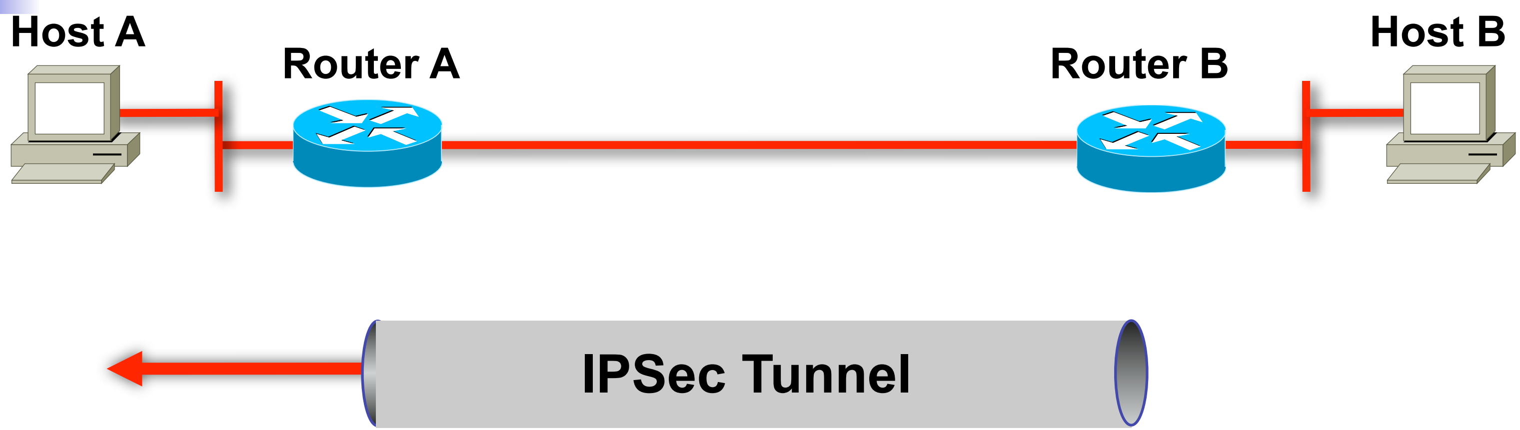
- Negotiates IPSec SA parameters protected by an existing IKE SA
- Establishes IPSec security associations
- Periodically renegotiates IPSec SAs to ensure security
- Optionally performs an additional Diffie-Hellman exchange
- If perfect forward secrecy is specified, a new Diffie-Hellman exchange is performed with each quick mode.

Step 4—IPSec Encrypted Tunnel



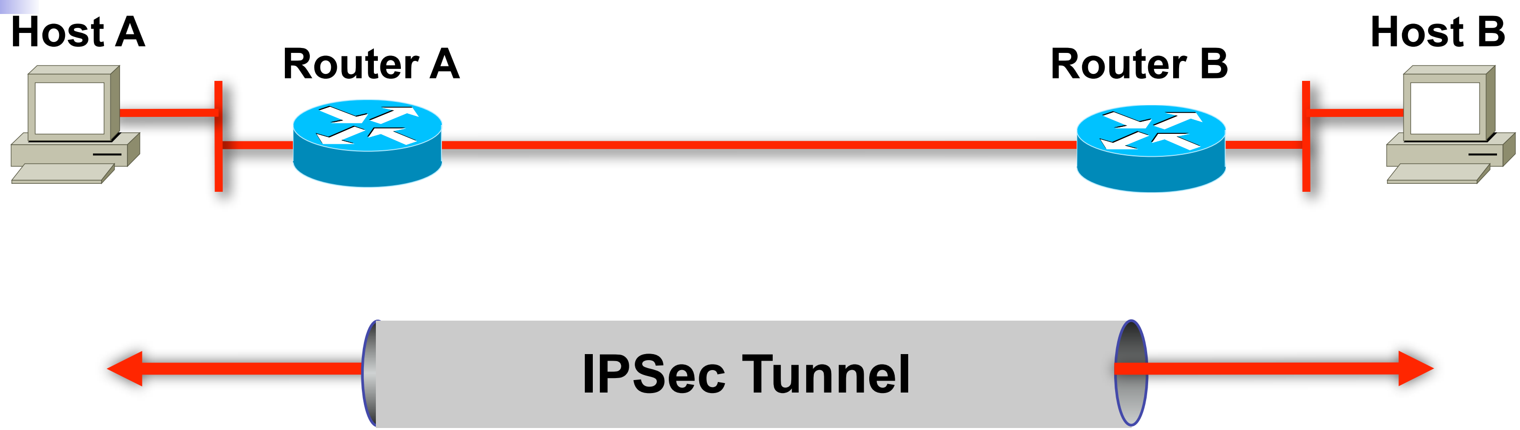
- Information is exchanged via IPSec tunnel.
- Packets are encrypted and decrypted.
- Uses encryption specified in IPSec SA.

Step 4—IPSec Encrypted Tunnel



- Information is exchanged via IPSec tunnel.
- Packets are encrypted and decrypted.
- Uses encryption specified in IPSec SA.

Step 4—IPSec Encrypted Tunnel



- Information is exchanged via IPSec tunnel.
- Packets are encrypted and decrypted.
- Uses encryption specified in IPSec SA.

Step 5—Tunnel Termination

- Tunnel is terminated by
  - TCP session termination
  - SA lifetime timeout
  - Packet counter exceeded
- Removes IPSec SA

Step 5—Tunnel Termination

- Tunnel is terminated by
  - TCP session termination
  - SA lifetime timeout
  - Packet counter exceeded
- Removes IPSec SA

Step 5—Tunnel Termination

- Tunnel is terminated by
  - TCP session termination
  - SA lifetime timeout
  - Packet counter exceeded
- Removes IPSec SA

Security Association

- Agreement between two entities on method to communicate securely
- IPSec SA is unidirectional
- Two-way communication consists of two SAs

IPSec SA

Destination Address	**192.168.2.1**
Security Parameter Index (SPI)	**7A390BC1**
IPSec Transform	**AH, HMAC-MD5**
Key	**7572CA49F7632946**
Additional SA Attributes (for example, lifetime)	**One Day or 100MB**

SA Parameter Example for Cisco Routers

R1 **R2**

R1:
- outbound esp sas:
- spi: 0x1B781456 (460854358)
- transform: esp-des ,
- in use settings ={Tunnel, }
- slot: 0, conn id: 18, crypto map:mymap

R2:
- inbound esp sas:
- spi: 0x1B781456(460854358)
- transform: esp-des ,
- in use settings ={Tunnel, }
- slot: 0, conn id: 18, crypto map:mymap
- sa timing: (k/sec)
- replay detection support: N

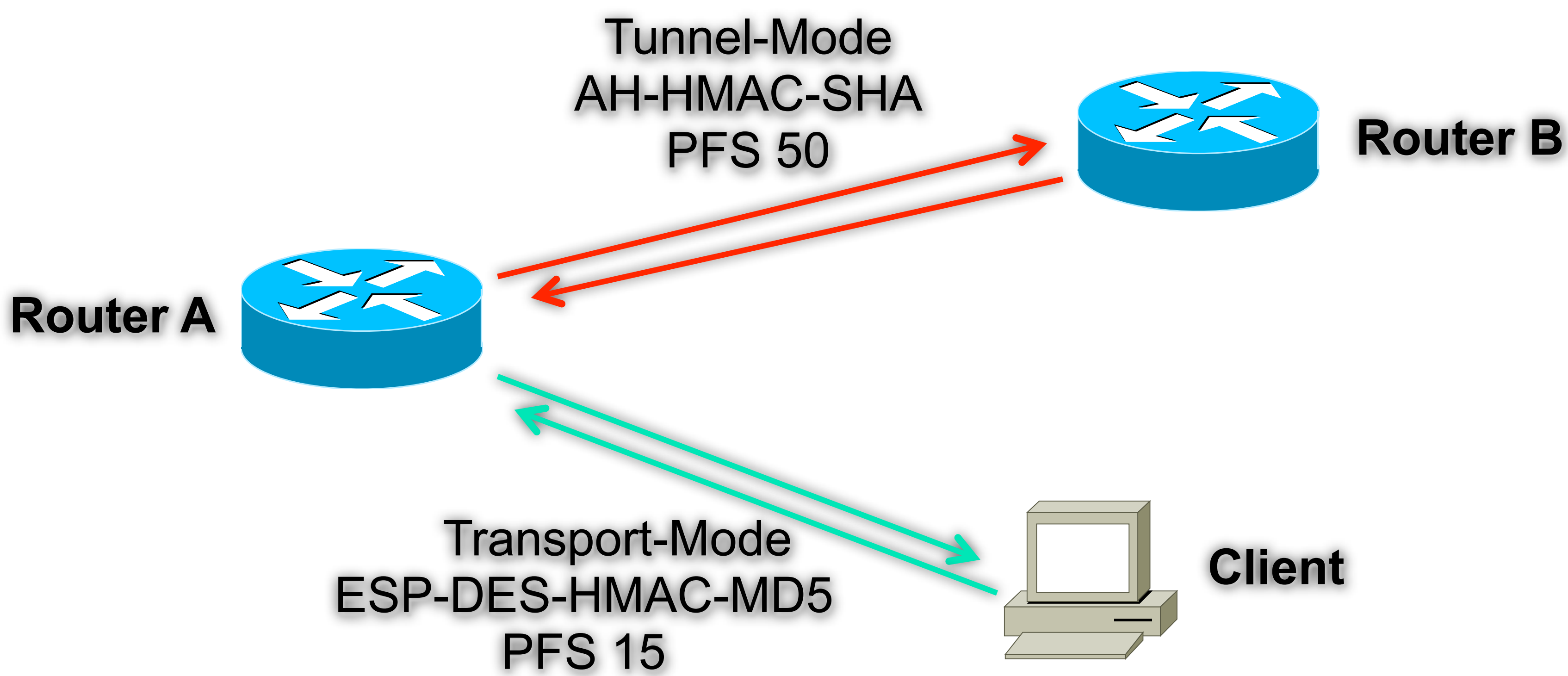
R1:

inbound esp sas:
spi: 0x8AE1C9C(145628316)
transform: esp-des ,
in use settings ={Tunnel, }
slot: 0, conn id: 17,
crypto map:mymap
sa timing: (k/sec)
replay detection support: N

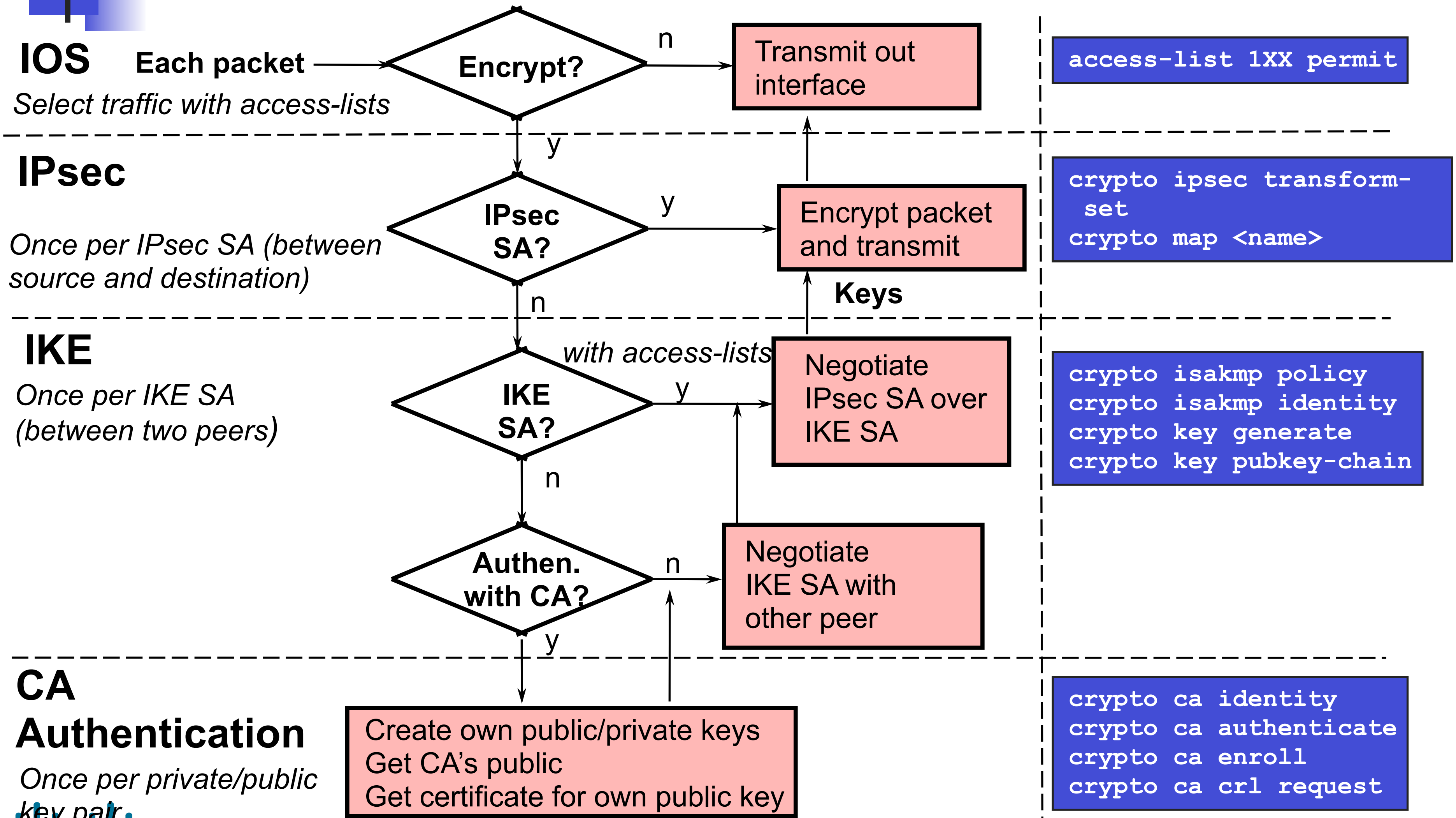
R2:

outbound esp sas:
spi: 0x8AE1C9C(145628316)
transform: esp-des ,
in use settings ={Tunnel, }
slot: 0, conn id: 17,
crypto map:mymap
sa timing: (k/sec)
replay detection support: N

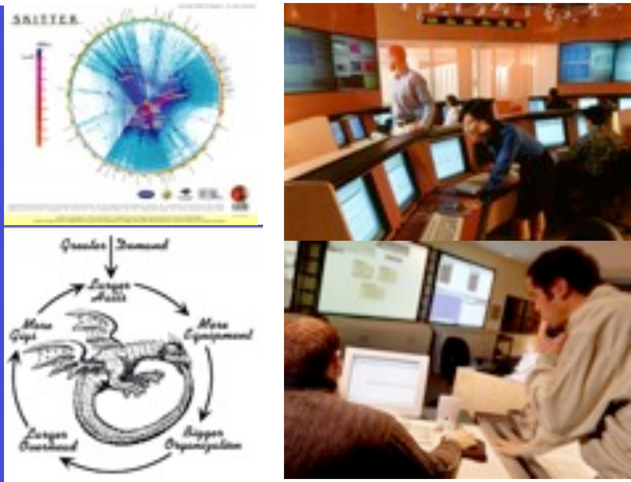
SAs Enable Your Chosen Policy



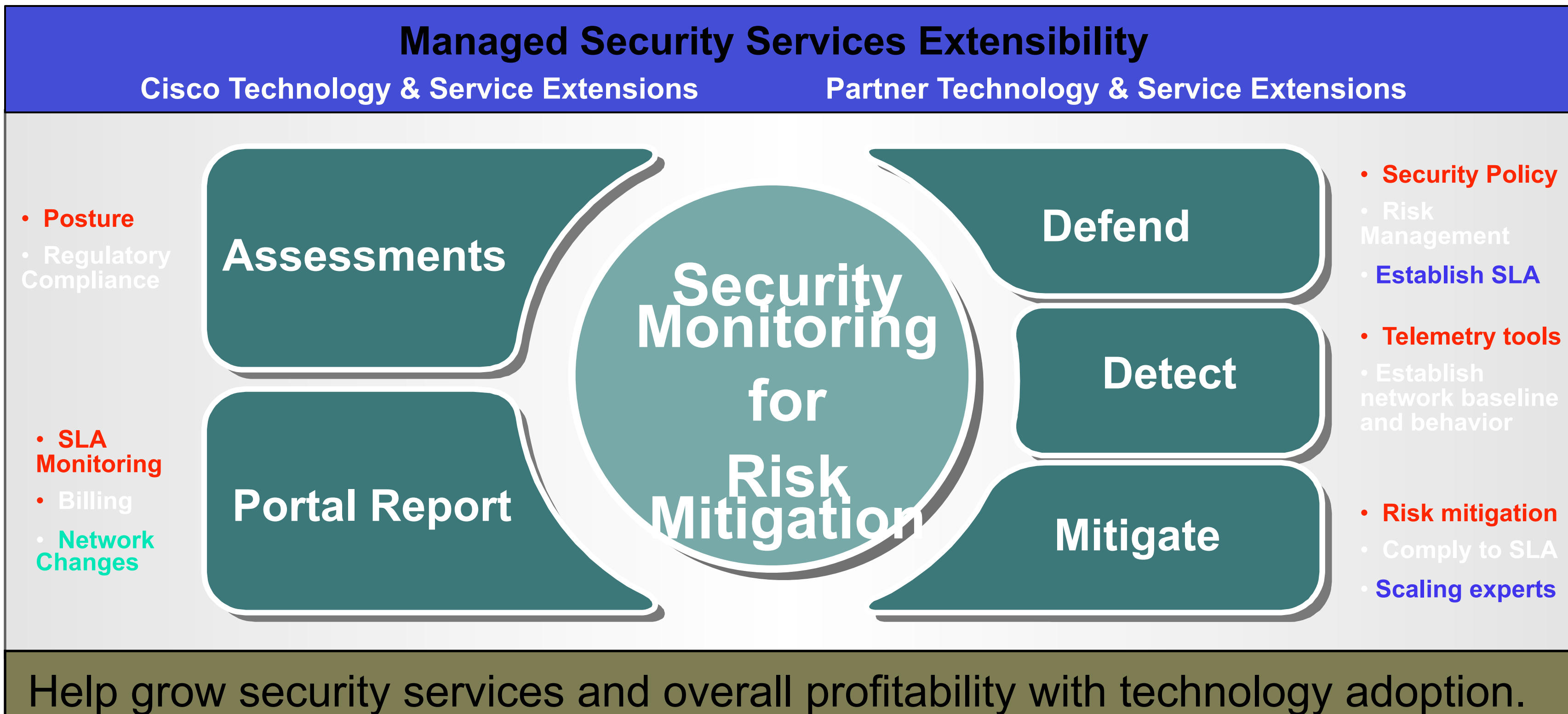
IKE and IPSec Flowchart for Cisco Routers



Managed Security Services



Managed Security Services Architecture



Managed Security Services Architecture

ness
Ass

Managed Security Services Extensibility

Cisco Technology & Service Extensions | **Partner Technology & Service Extensions**

- Posture
- Regulatory Compliance

Assessments

- SLA Monitoring
- Billing
- Network Changes

Portal Report

Security Monitoring for Risk Mitigation

Defend

- Security Policy
- Risk Management
- Establish SLA

Detect

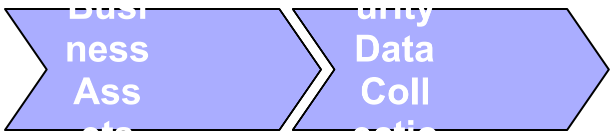
- Telemetry tools
- Establish network baseline and behavior

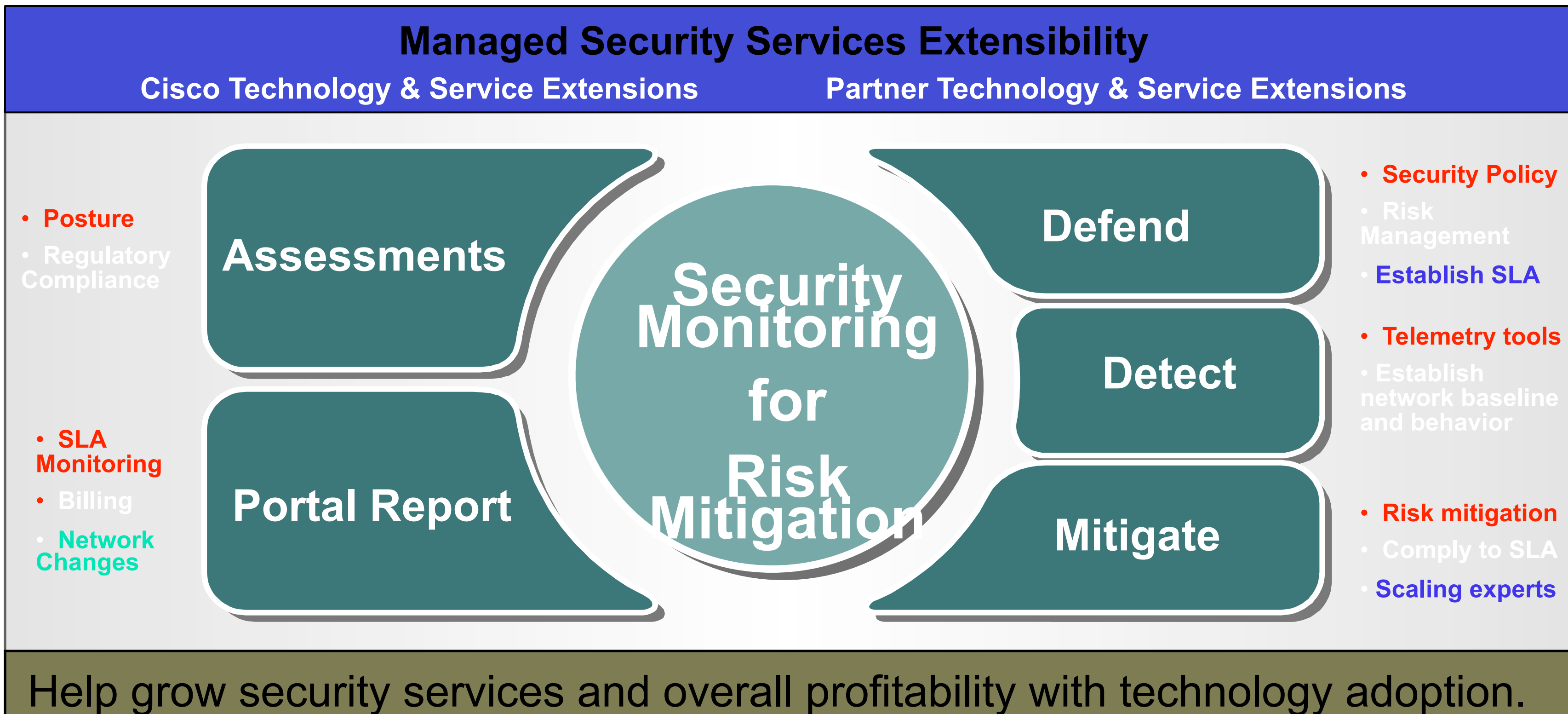
Mitigate

- Risk mitigation
- Comply to SLA
- Scaling experts

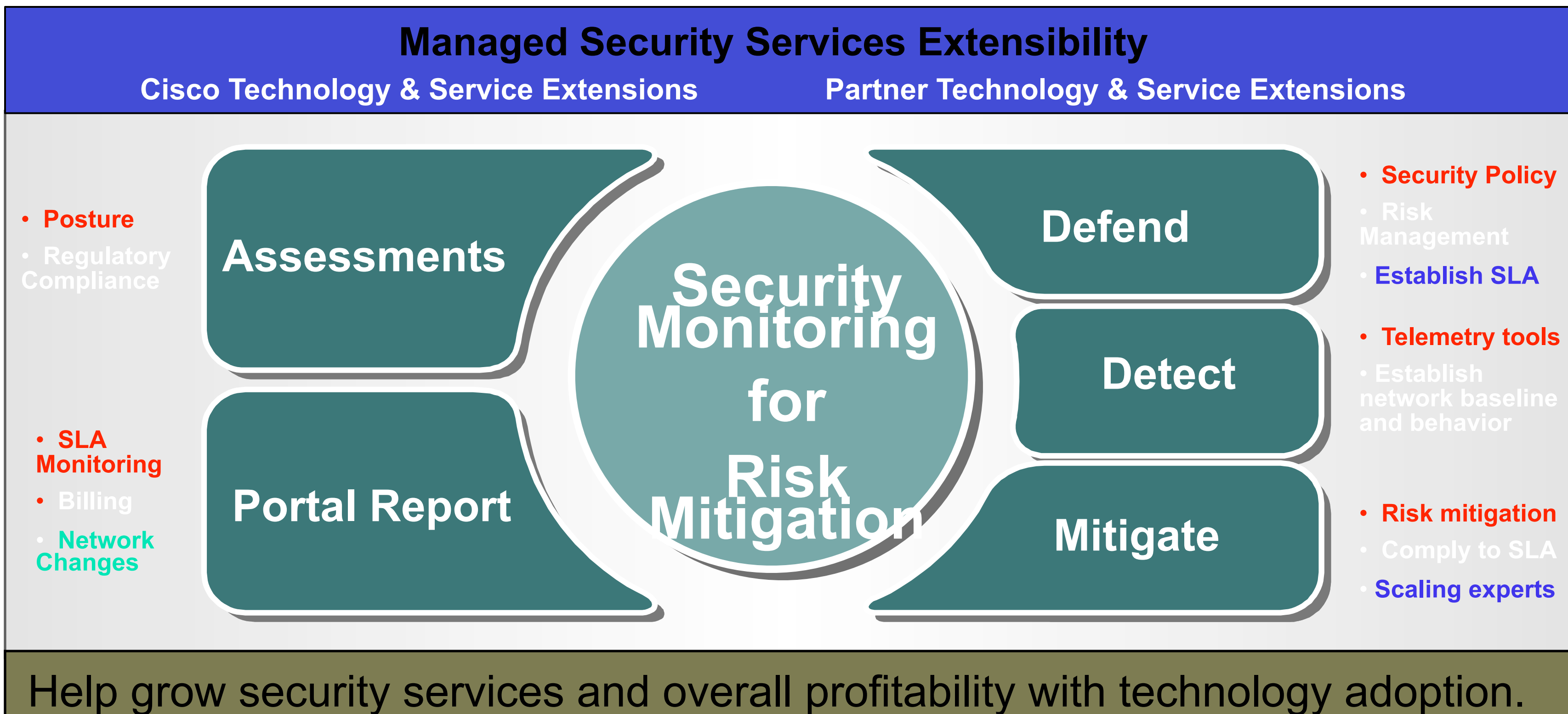
Help grow security services and overall profitability with technology adoption.

Managed Security Services Architecture



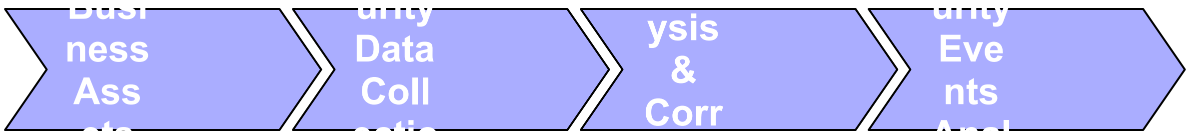


Managed Security Services Architecture



Cisco · Alcatel·Lucent · Juniper Networks

Managed Security Services Architecture



Managed Security Services Extensibility

Cisco Technology & Service Extensions | **Partner Technology & Service Extensions**

- Posture
- Regulatory Compliance

- SLA Monitoring
- Billing
- Network Changes

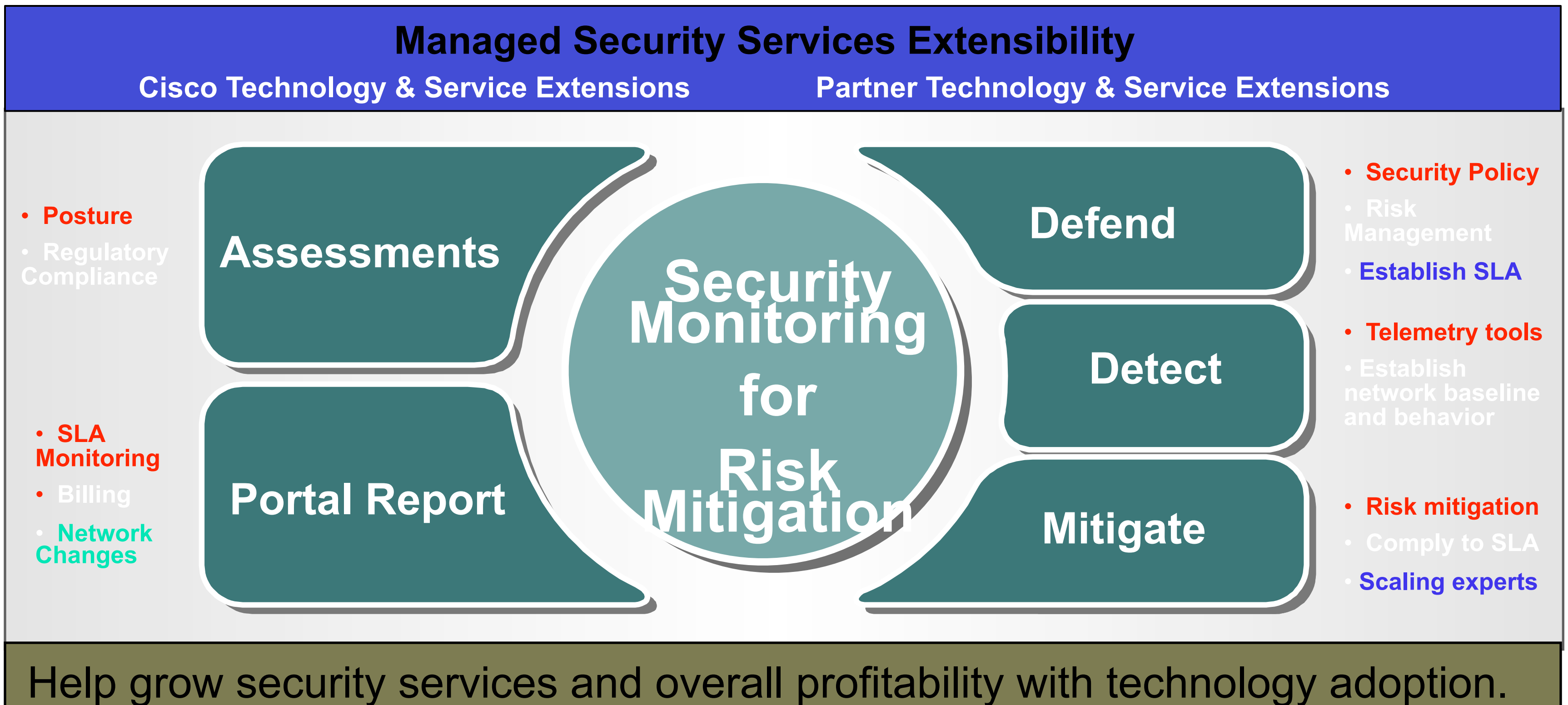
- Security Policy
- Risk Management
- Establish SLA

- Telemetry tools
- Establish network baseline and behavior

- Risk mitigation
- Comply to SLA
- Scaling experts

Help grow security services and overall profitability with technology adoption.

Managed Security Services Architecture



Managed Security Services Architecture

- ness Ass
- Data Coll
- ysis & Corr
- Eve nts
- ert Res pon
- Com munic ate

Managed Security Services Extensibility

Cisco Technology & Service Extensions | **Partner Technology & Service Extensions**

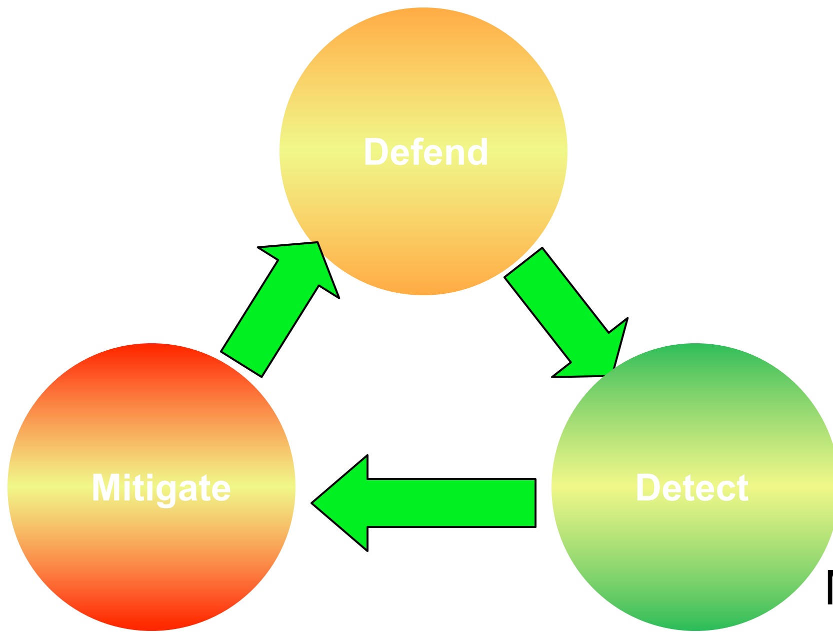
- Posture
- Regulatory Compliance
- SLA Monitoring
- Billing
- Network Changes

- Security Policy
- Risk Management
- Establish SLA
- Telemetry tools
- Establish network baseline and behavior
- Risk mitigation
- Comply to SLA
- Scaling experts

Help grow security services and overall profitability with technology adoption.

Critical Security Capabilities of MSS Offerings

Security policy representation with risk mitigation plans



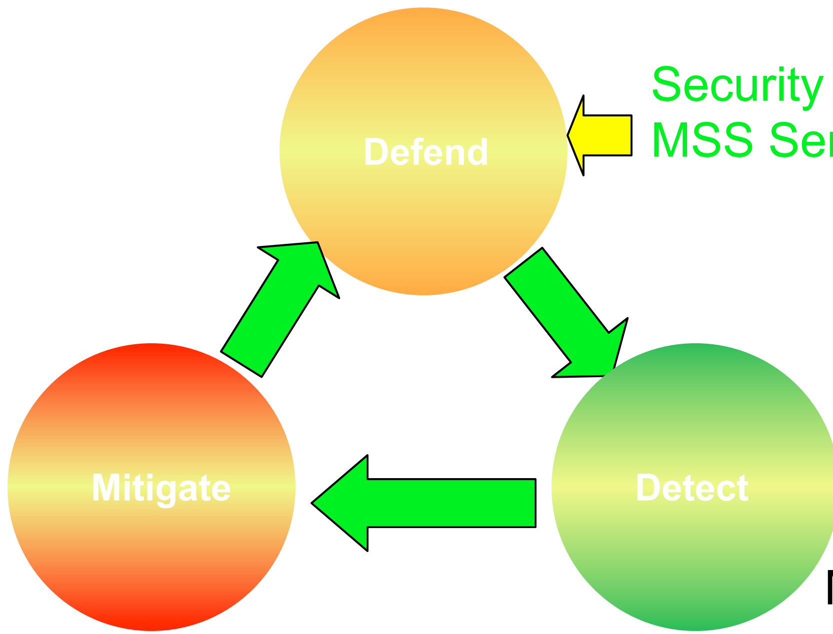
Mitigate risks to SLA defined in security policy

Monitoring for risk and detect threats

Critical Security Capabilities of MSS Offerings

Security policy representation with risk mitigation plans

Security Policy Establishment
MSS Service Level Agreements



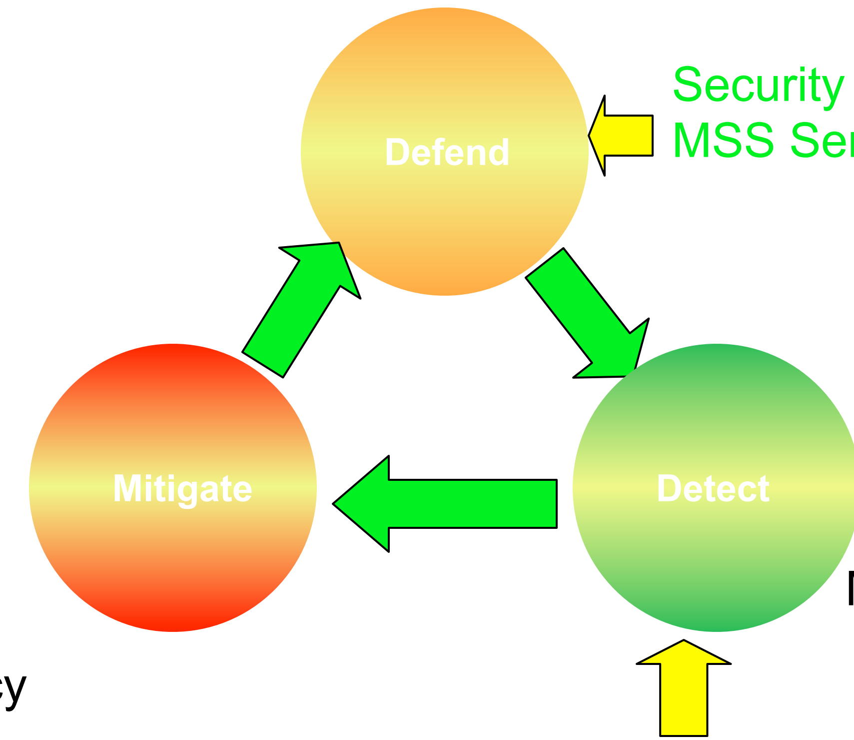
Monitoring for risk and detect threats

Mitigate risks to SLA defined in security policy

Critical Security Capabilities of MSS Offerings

Security policy representation with risk mitigation plans

Security Policy Establishment
MSS Service Level Agreements



Monitoring for risk and detect threats

Mitigate risks to SLA defined in security policy

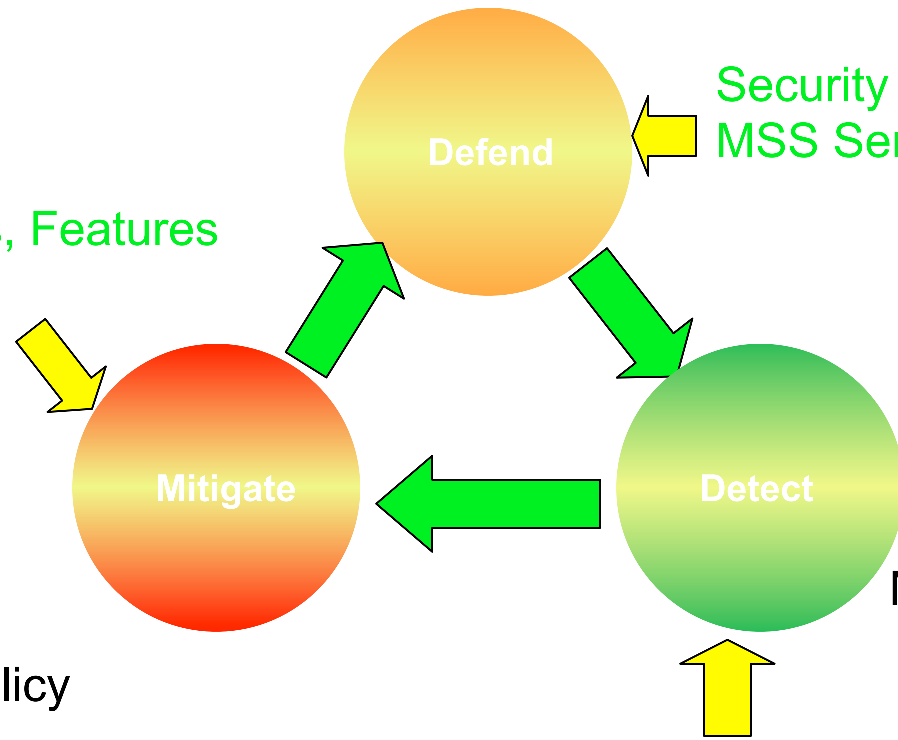
Network Telemetry
Data Analysis Intelligence

Critical Security Capabilities of MSS Offerings

Security policy representation with risk mitigation plans

Security Policy Establishment
MSS Service Level Agreements

Technology, Products, Features
Process for Actions



Monitoring for risk and detect threats

Mitigate risks to SLA defined in security policy

Network Telemetry
Data Analysis Intelligence

MSS Offerings Strategy and Roadmap

MSS Offerings Strategy and Roadmap

Managed Security Services - IDS

Managed Security Service - VPN

Managed Security Services – Firewall

Managed Router Service
Connecting to the ‘I’

Secure Access Services

Revenue And Customer Retention

Alcatel·Lucent

JUNIPER NETWORKS

MSS Offerings Strategy and Roadmap

Enhanced Security Services and Options

Managed Content Security
Managed End Point Protection – NAC/CSA
Managed Security Services – IPS
Enhanced Security Services and Options

Revenue And Customer Retention

Managed Security Services - IDS
Managed Security Service - VPN
Managed Security Services – Firewall
Managed Router Service Connecting to the 'I'

Secure Access Services

MSS Offerings Strategy and Roadmap

Revenue And Customer Retention

Enhanced Security Services and Options

Managed Content Security

Managed End Point Protection – NAC/CSA

Managed Security Services – IPS

Enhanced Security Services and Options

Service Continuum

Managed Security Services - IDS

Managed Security Service - VPN

Managed Security Services – Firewall

Managed Router Service Connecting to the 'I'

Secure Access Services

Managed Firewall Service

Pain Points for Customers

- Protect internal and external networks
- Protect against embedded vulnerabilities
- Allow secure access to users

Service Opportunity for SP

- Value-add on transport provision
- ISR run firewall with hardware acceleration

Managed Firewall Service

Pain Points for Customers

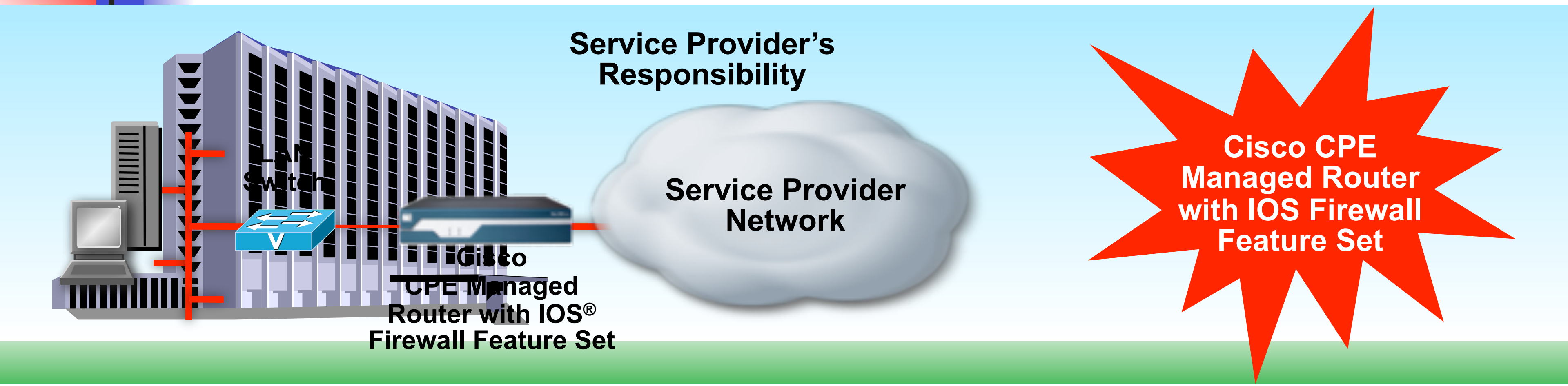
- Protect internal and external networks
- Protect against embedded vulnerabilities
- Allow secure access to users

Service Opportunity for SP

- Value-add on transport provision
- ISR run firewall with hardware acceleration

Integrated firewall results in operational savings

Managed Firewall Service In Details

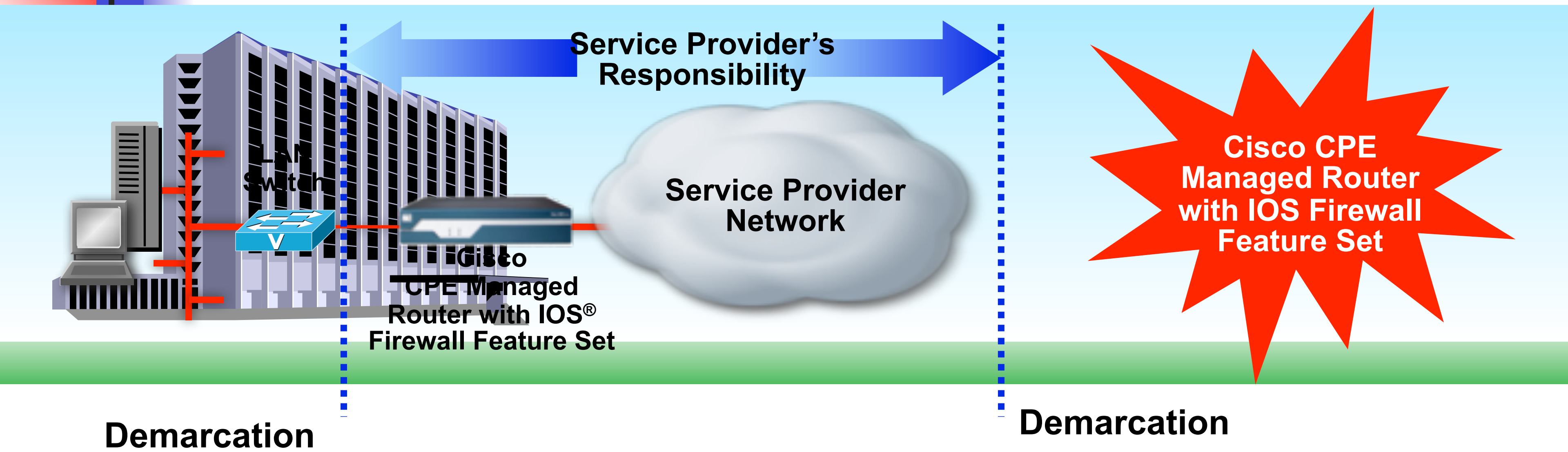


Demarcation

Demarcation

- **Managed router service allows remote configuration**
- **SP enables firewall on a managed router**
- **Basic firewall allows split tunneling and dynamic site-to-site connections**
- **Advanced firewall allows application filtering to comply with security policies**
- **Provide Managed IOS Firewall Service without truck-roll**
- **Obtain new service revenue by already deployed and managed CPE**

Managed Firewall Service In Details



- **Managed router service allows remote configuration**
- **SP enables firewall on a managed router**
- **Basic firewall allows split tunneling and dynamic site-to-site connections**
- **Advanced firewall allows application filtering to comply with security policies**
- **Provide Managed IOS Firewall Service without truck-roll**
- **Obtain new service revenue by already deployed and managed CPE**

Managed VPN Service

Pain Points for Customers

- Secure scalable connectivity
- Address regulatory requirements

Service Opportunity for SP

- Extend network to new users and locations
- Remote management ensure optimal performance and scale

Managed VPN Service

Integrated VPN results in new users and locations

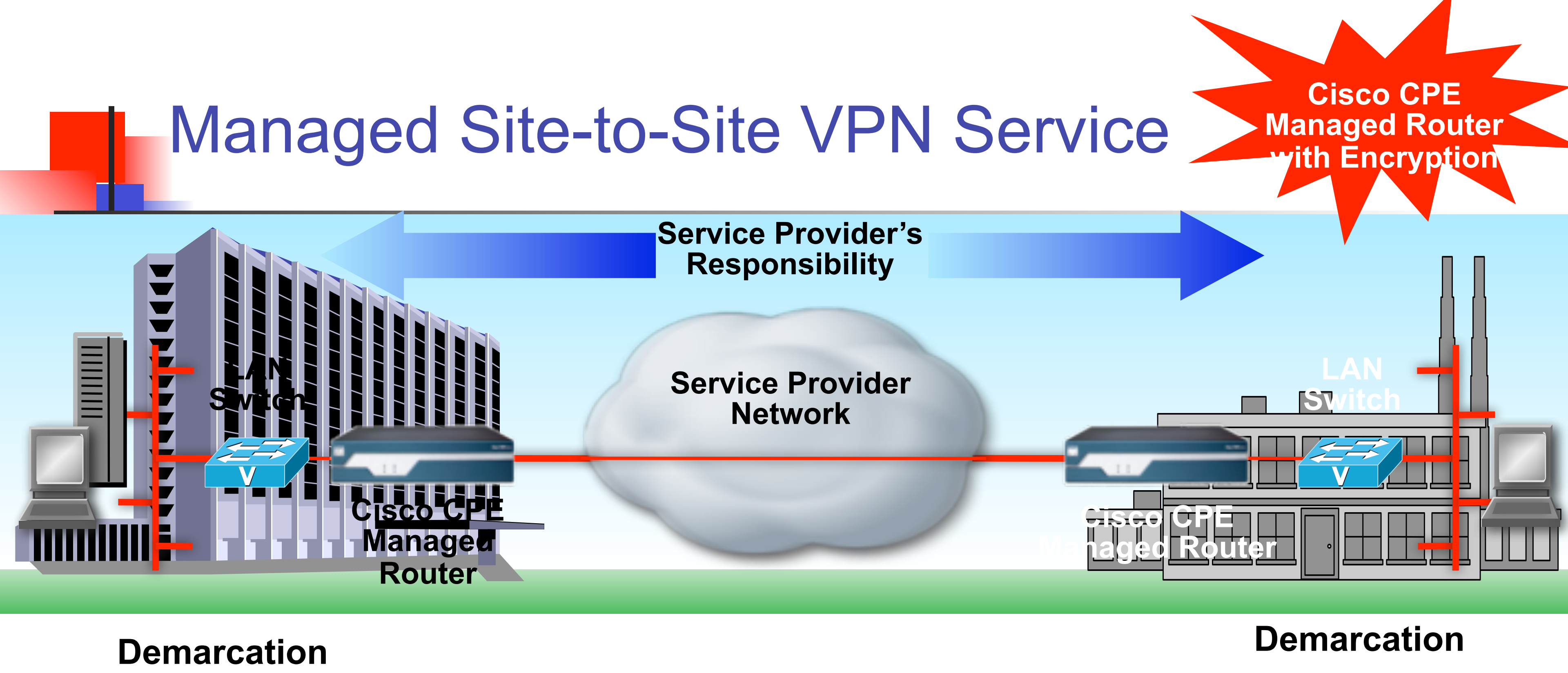
Pain Points for Customers

- Secure scalable connectivity
- Address regulatory requirements

Service Opportunity for SP

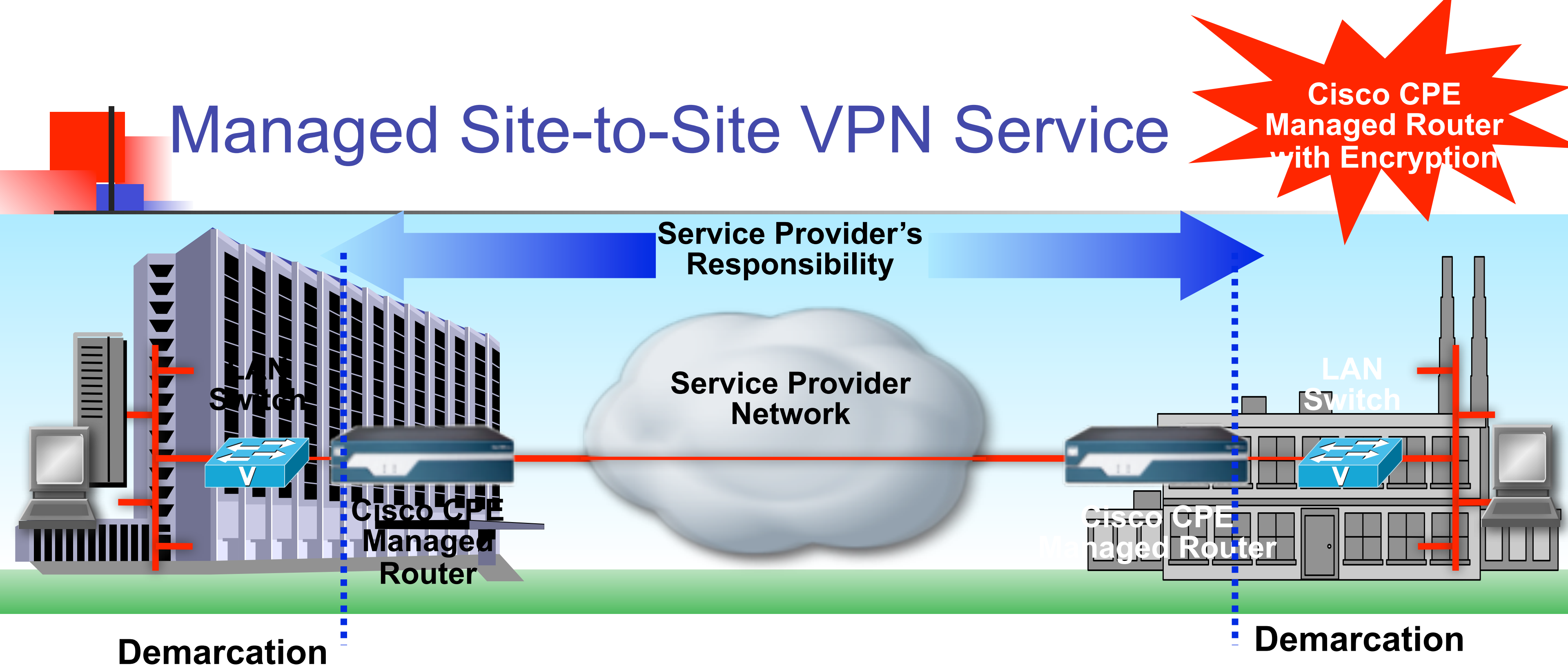
- Extend network to new users and locations
- Remote management ensure optimal performance and scale

Managed Site-to-Site VPN Service



- Managed router service allows remote configuration
- SP enables Site-to-Site MPLS or IPSec VPN features in Cisco Router
- Extend support for VPN Acceleration from SP to customer premise through AES wide-key support in both
- Provide Managed Site-to-Site VPN Service without truck-roll
- Obtain new service revenue from already deployed and managed CPE

Managed Site-to-Site VPN Service



- Managed router service allows remote configuration
- SP enables Site-to-Site MPLS or IPSec VPN features in Cisco Router
- Extend support for VPN Acceleration from SP to customer premise through AES wide-key support in both
- Provide Managed Site-to-Site VPN Service without truck-roll
- Obtain new service revenue from already deployed and managed CPE

Managed IDS/IPS Service

Pain Points for Customers

- Attack mitigation and threat prevention distributed at all network entry points
- Identify, classify and stop malicious traffic in real-time

Service Opportunity for SP

- Add value by real time monitoring and threat mitigation for customers
- Create loyalty by customizing solutions for unique customer security policy
- Provide remote management for security service and provide comprehensive reporting on the security events

Managed IDS/IPS Service

Integrated IPS results in increased network visibility

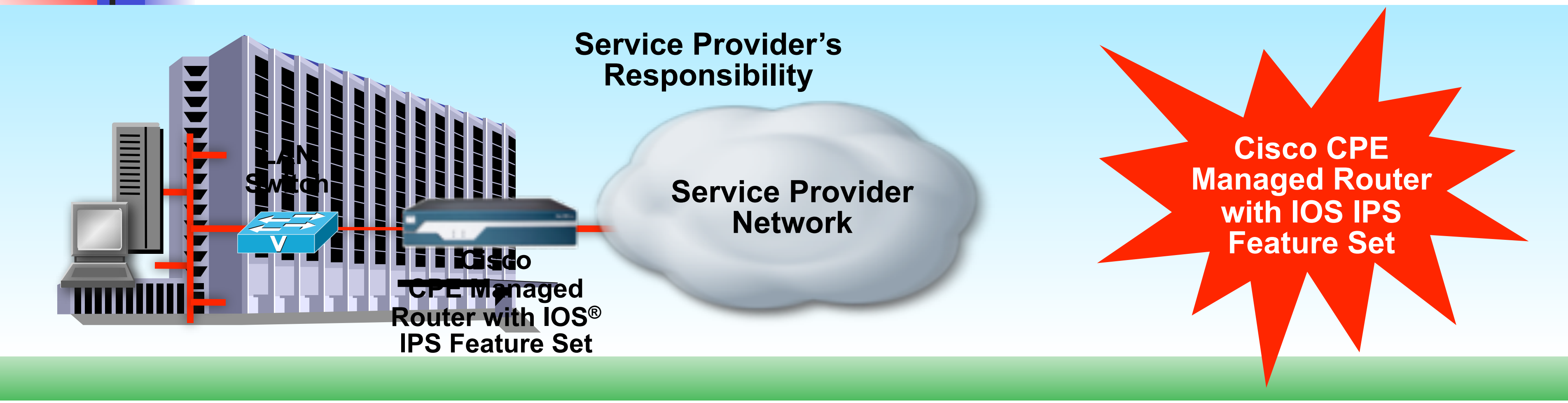
Pain Points for Customers

- Attack mitigation and threat prevention distributed at all network entry points
- Identify, classify and stop malicious traffic in real-time

Service Opportunity for SP

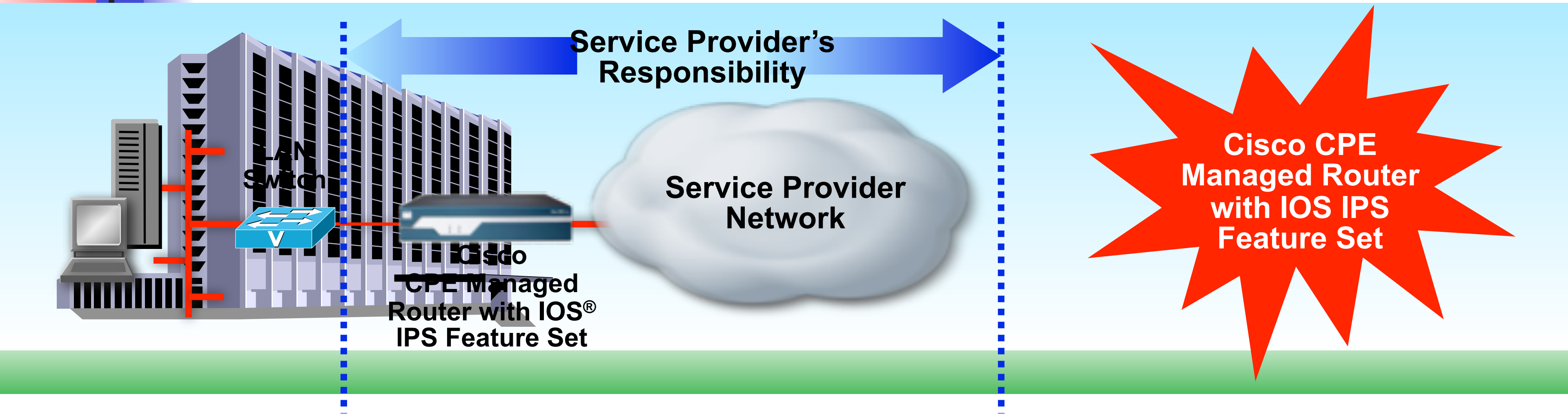
- Add value by real time monitoring and threat mitigation for customers
- Create loyalty by customizing solutions for unique customer security policy
- Provide remote management for security service and provide comprehensive reporting on the security events

Managed IDS/IPS Service In Details



- SP configures IOS IDS/IPS Feature Set in Cisco Router by configuration management feature of Managed Router Service
- Provide Managed IOS Firewall Service and Intrusion Protection Service without truck-roll
- Provide Inline IPS option- customizable signatures can be dynamically loaded
- Obtain new service revenue by already deployed and managed CPE

Managed IDS/IPS Service In Details



- **SP configures IOS IDS/IPS Feature Set in Cisco Router by configuration management feature of Managed Router Service**
- **Provide Managed IOS Firewall Service and Intrusion Protection Service without truck-roll**
- **Provide Inline IPS option- customizable signatures can be dynamically loaded**
- **Obtain new service revenue by already deployed and managed CPE**

Managed Endpoint Protection Service

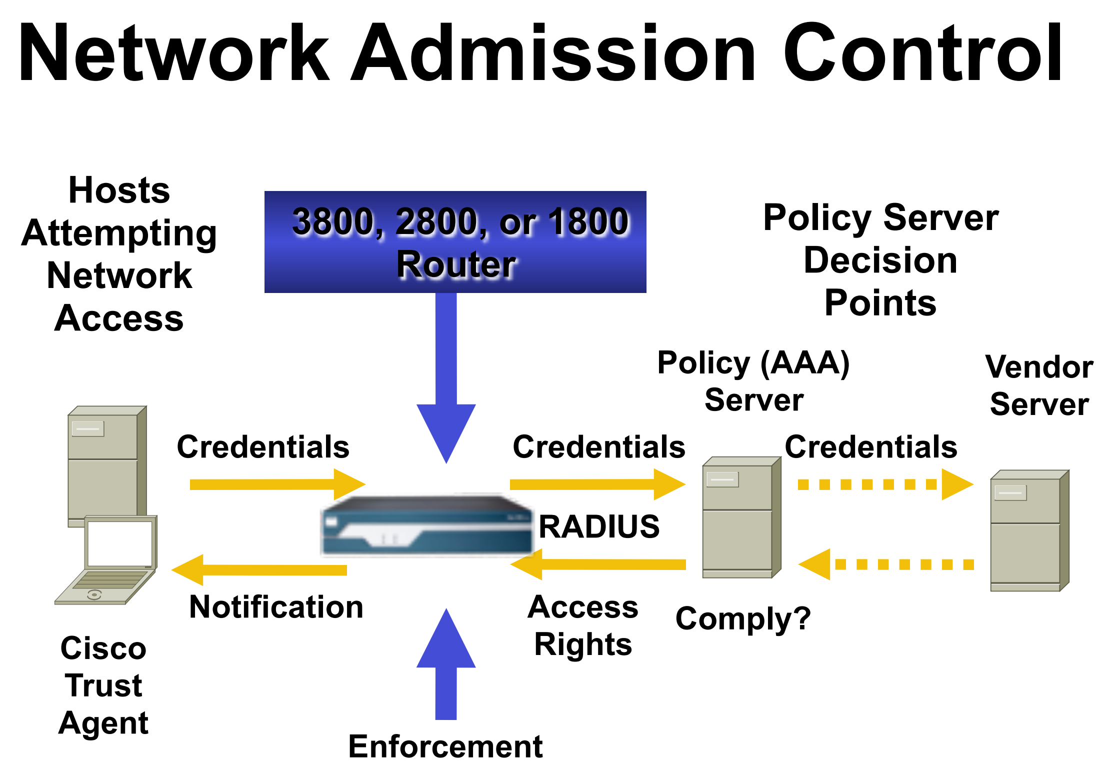
Pain Points for Customers

- Protection Beyond the Perimeter
- Detect and mitigate

Service Opportunity for SP

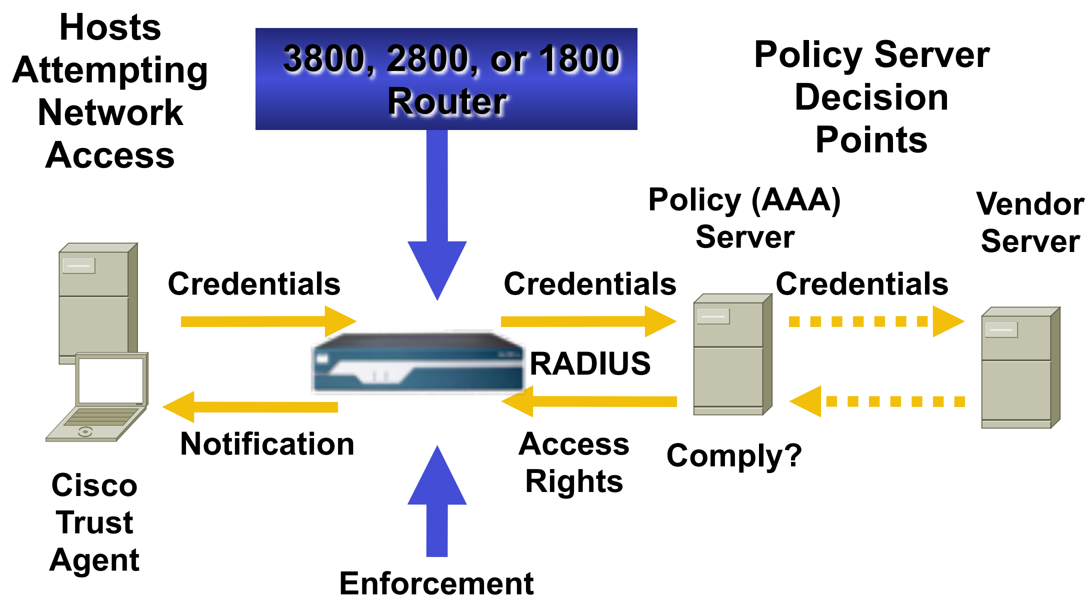
- Emerging service opportunity to further penetrate customer network and LAN environment
- Leverage the network to intelligently enforce access privileges based on endpoint security posture

Introducing NAC



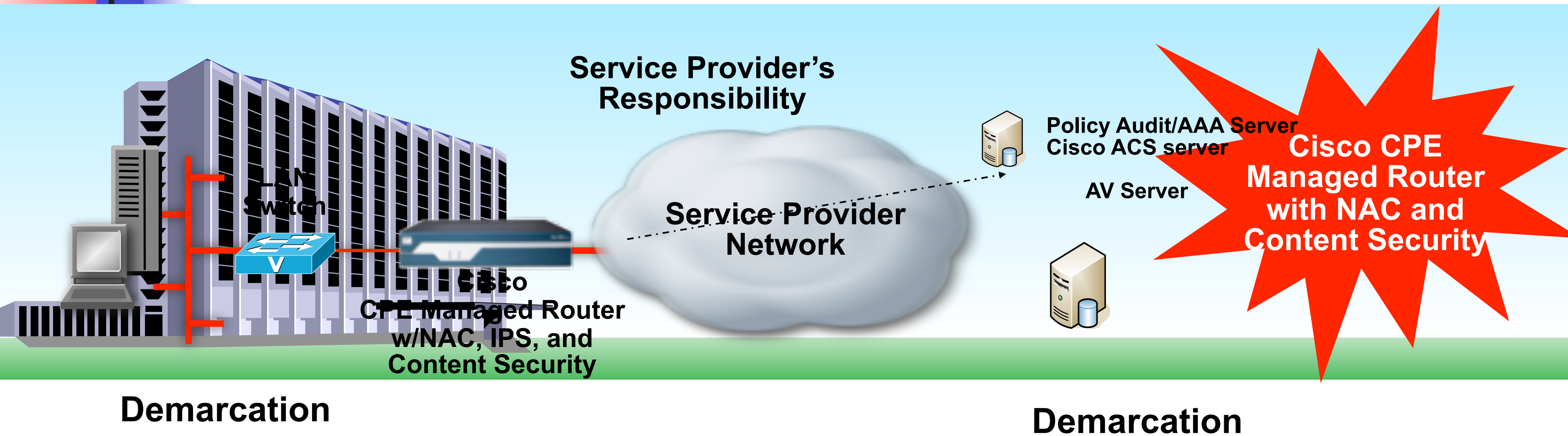
Introducing NAC

Network Admission Control



Coalition of industry leading partners

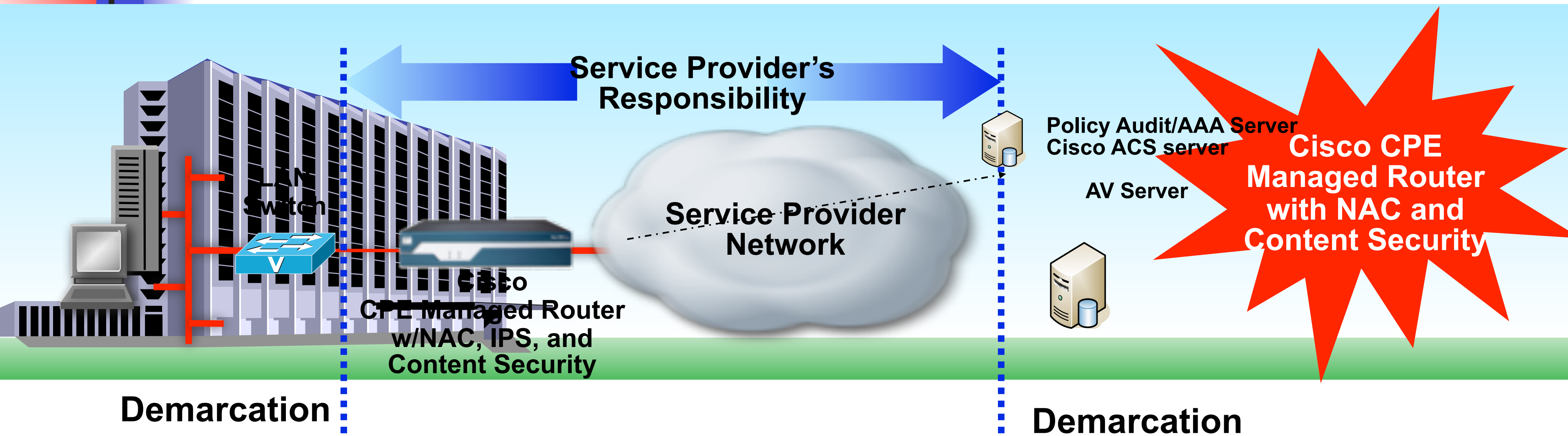
Managed Endpoint Protection Service In Details



Cisco Network Admission Control technology within CPE Router

- ISR as a CPE is a security policy enforcement point and provides visibility to the network behavior
- Managed ISR supports access control and identity
- Network collaborates with applications
- Layer 2 and Layer 3 collaboration provides in per user level policy management
- Facilitate security policy audit and compliance
- NAC is an enhanced service so obtain new service revenue by already deployed and managed CPE

Managed Endpoint Protection Service In Details



Cisco Network Admission Control technology within CPE Router

- ISR as a CPE is a security policy enforcement point and provides visibility to the network behavior
- Managed ISR supports access control and identity
- Network collaborates with applications
- Layer 2 and Layer 3 collaboration provides in per user level policy management
- Facilitate security policy audit and compliance
- NAC is an enhanced service so obtain new service revenue by already deployed and managed CPE

Managed Services Lab

Conclusions

- What is the first step after the workshop?
  - Be part of the community
  - Consider iNOC/DBA
  - Utilize NetFlow / SNMP / SysLog and Open Source tools for Telemetry
  - Deploy RTBH for Source and Destination based dropping
- What Managed Security Services your organization can offer?
  - Managed VPN Services: MPLS and IPSec
  - Managed Firewall Services

Q and A

APRICOT 2006 ISP Security L3VPN Lab