IP Multicast Security

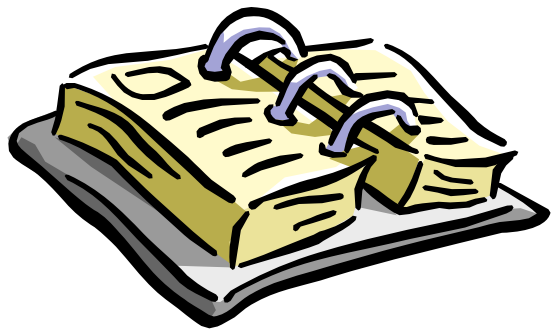
RST-2262

Objectives

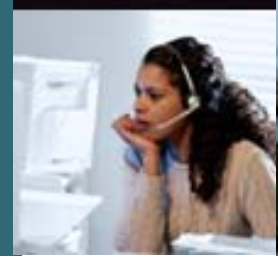
- **Not a cookbook of recipes!**
- **Learn what makes multicast different from unicast**
- **Learn about tools (router functions) available**
  - **Means and goal,**
  - **Tools developed for unicast**
  - **Multicast specific tools**
  - **With example stubs**

Agenda

- Introduction
- Control Plane Security
- Access Control
- Admission Control
- AAA
- Firewalls
- IPsec



Introduction



Security Goals …
(Why We Want “Security”)

- **Keep network running under**

  Mis-configuration, Malfunction, Attacks

- **Manage resources**
- **Control multicast application/service**

  Sender / Receiver

  Authorization / Subscription

- **Account for**

  Resource utilization, service participation

No simple 1:1 mapping between goals and means.
Wide range of tools in products.

… and Means
What “Tools” Provide Security

- **Service Control and Enforcement**
- **AAA**

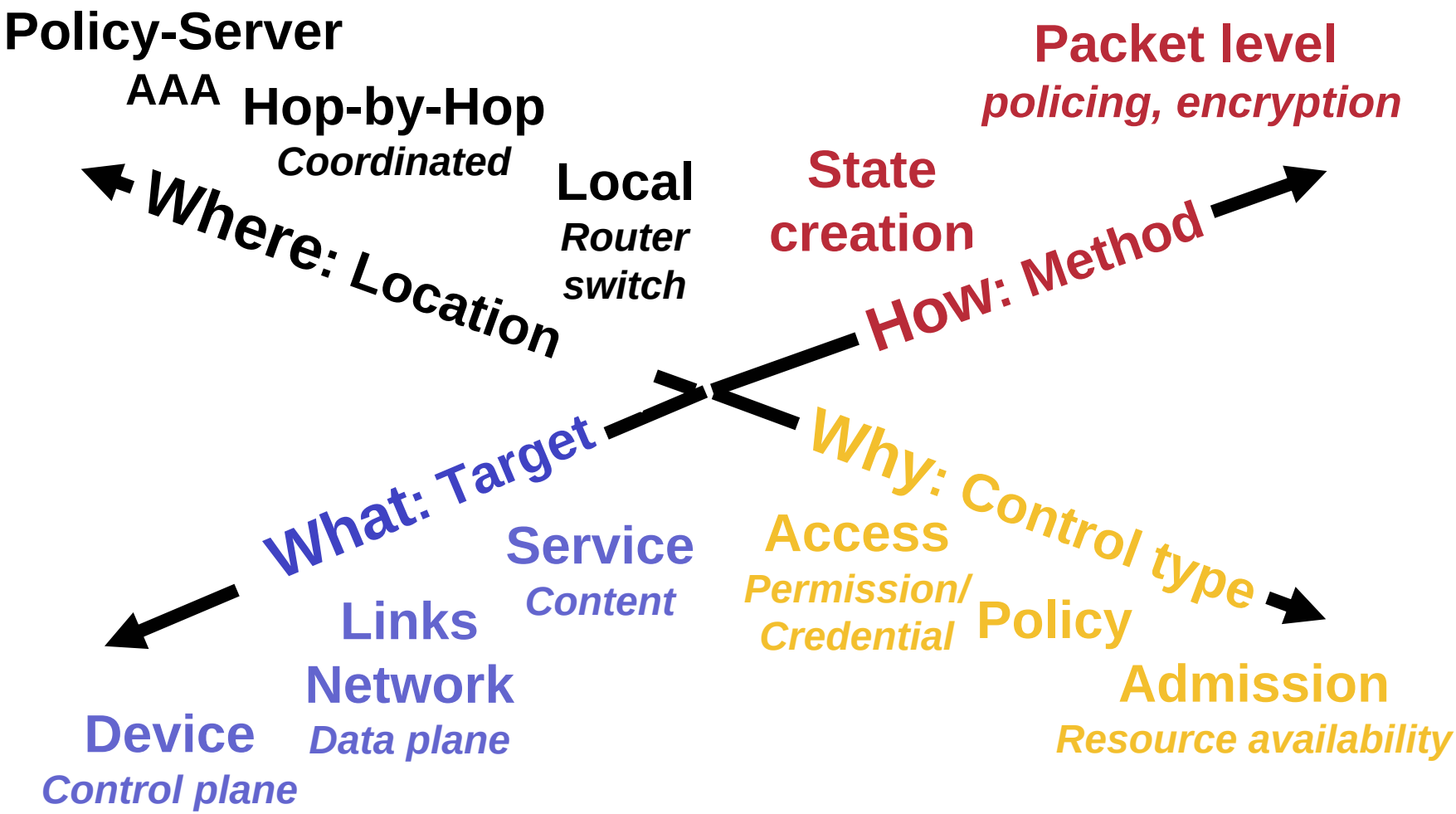
  Authentication, Authorization and Accounting
- **“Privacy”**

  Closed user groups (scopes), VPN / VRF
- **NAT ?**

  not security!

  Just collocated in firewalls

Multicast Control and Enforcement



Multicast / Unicast Security Comparison

- **Network**
  - Replication
  - Per application state
  - Receiver built trees
  - Scoped addresses
- **Application**
  - Typical not “TCP like”
  - Unidirectional or multidirectional
- **Protocols**
  - Similar
  - Common use of link scope multicast

Multicast / Unicast Security Comparison

Think Multicast !!!

… if you want multicast security…

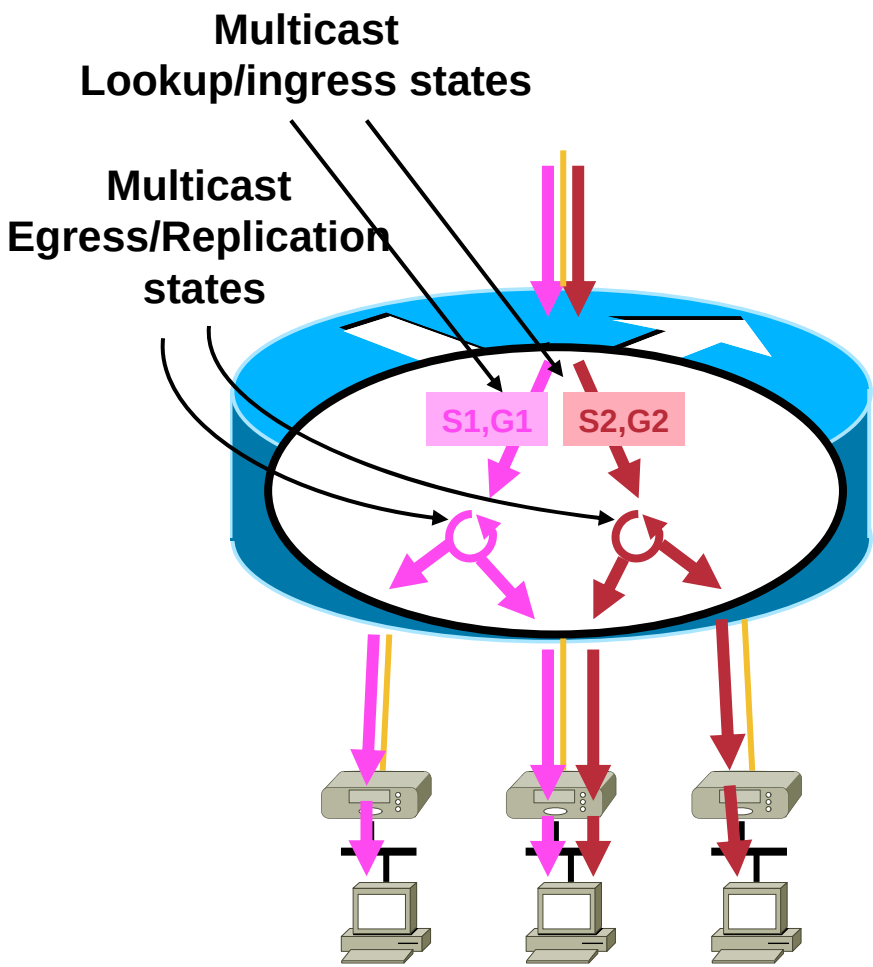
CAN DO MORE / MUST DO MORE

Multicast vs. Unicast
Per-Application State

- **Unicast:**
  - State grows when network topology grows.
  - CPU is active when network topology changes.
    - No impact by user activity
    - Design network (only) for bandwidth
    - Topology change triggered activity
- **Multicast:**
  - All of unicast
  - State grows when user starts application
  - CPU active when application state changes
    - Design for number of application/sources

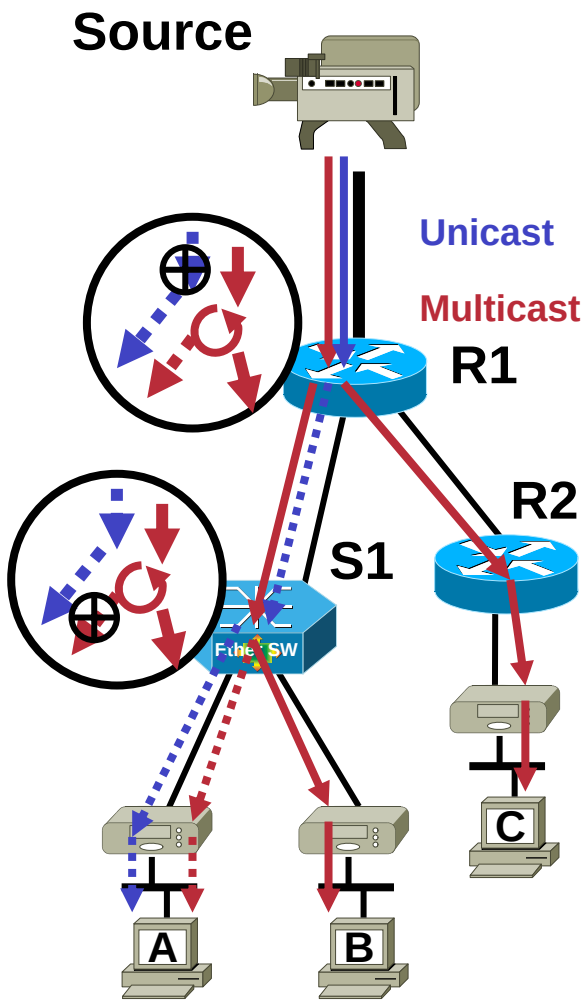
Multicast vs. Unicast
State and Replication in Routers and Switches

- “ingress” state
  - per application/sender
- “egress” state
  - per receiver branch
- HW limits: 5000 … >100,000
- SW limits: >> 100,000
- Throughput limits
  - Unicast: Ingress Packet Rate
  - Multicast: Egress Packet Rate
- ROUTERS AND SWITCHES



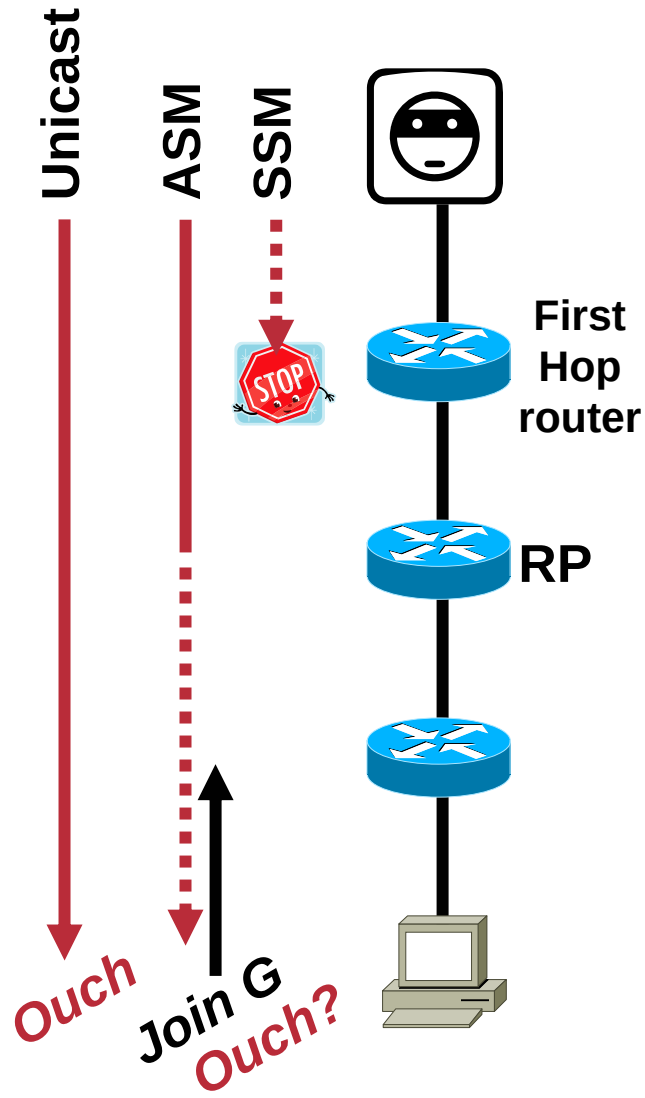
Multicast vs. Unicast
Replication in Routers and Switches

- **Example: Inhibit Source -> A traffic**
- **Unicast: can filter anywhere on path**
- **Multicast**
  - **Switch or router**
  - **Receiver:**
    - **MUST filter after last replication**
    - **Egress filtering!**
  - **Sources:**
    - **MUST filter before first replication**
    - **Ingress filtering!**



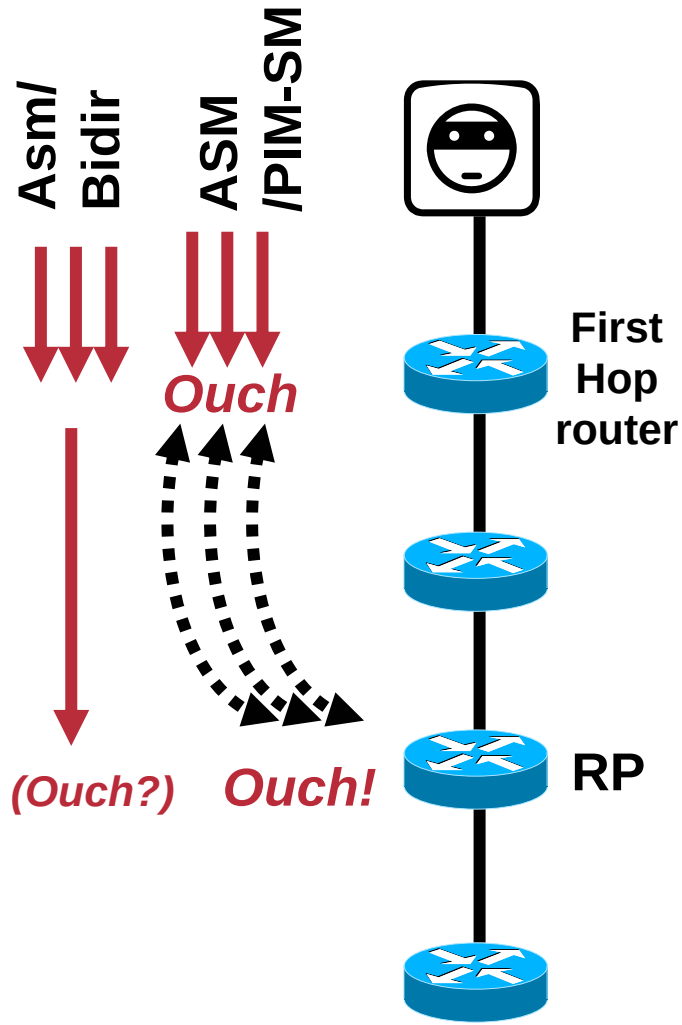
Multicast vs. Unicast
Receiver Side Explicit Join Based Traffic Forwarding (1)

- Attacks from sources to hosts:
- **Unicast: No implicit protection.**
  - **Main reason for Firewalls.**
- Multicast: implicit protection
- **ASM:**
  - **Sources can attack groups**
  - **No independent host attacks**
- **SSM:**
  - **No attacks by unwanted sources**
  - **Traffic stops at first-hop router**



Multicast vs. Unicast
Receiver Side Explicit Join Based Traffic Forwarding (2)

- **Attacks from sources to network:**
- **Even without receivers**
- **PIM-SM:**
  - **(S,G) and (*,G) on FHR and RP**
  - **State attack!**
- **Bidir-PIM:**
  - **No state attack – just traffic !**
    - **RP as attackable as unicast**
  - **(*,G/M) towards RP.**
  - **Note: IOS IPv4 multicast still creates (*,G) state due to legacy implementation (except cat6k/c7600).**



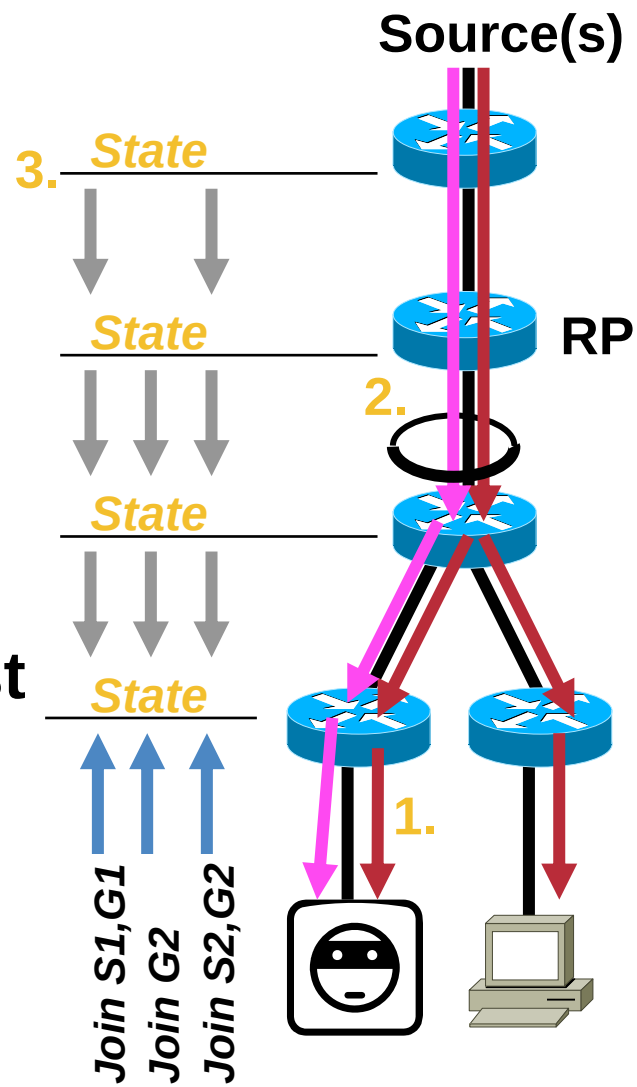
Multicast vs. Unicast
Receiver Side Explicit Join Based Traffic Forwarding (3)

- **Attacks from receivers!**

  Receivers create state

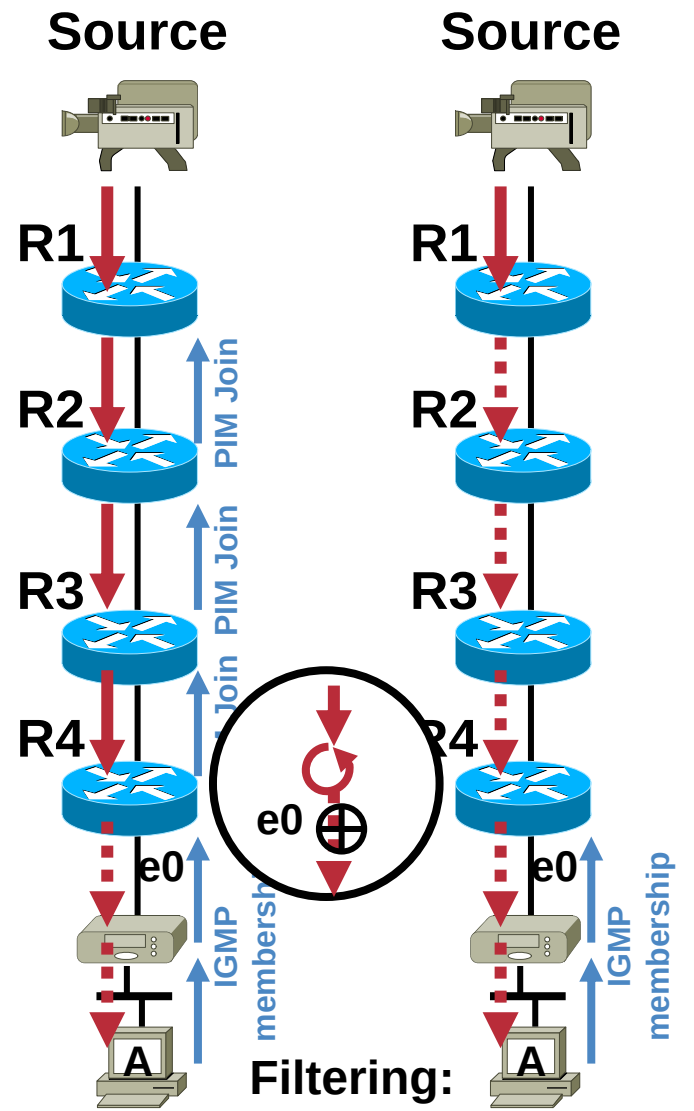
  No equivalent in unicast.

1. **Attack against content:**
   receive content unauthorized
2. **Attack against bandwidth:**
   Overload network bandwidth.
   Shared bandwidth: attack against other receivers
3. **Attack against routers/switches:**
   Overload state tables
   Increase convergence times



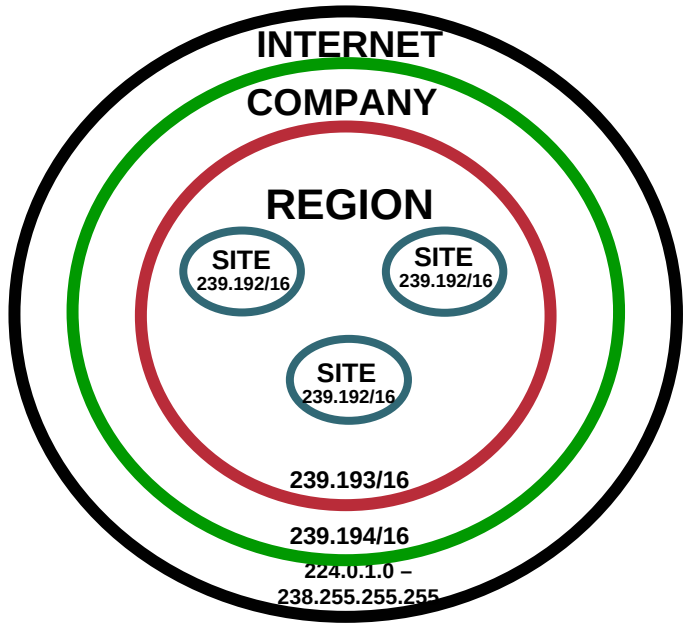
Multicast vs. Unicast
Receiver Side Explicit Join Based Traffic Forwarding (4)

- **Unicast**
  - **Can only (well) filter packets**
  - **“ip access-group <acl>”**
- **Multicast**
  - **Can stop traffic forwarding by filtering at control plan**
  - **Receiver side filtering stops traffic at source**
  - **“ip multicast boundary” …**



Multicast vs. Unicast
Scoped Addresses

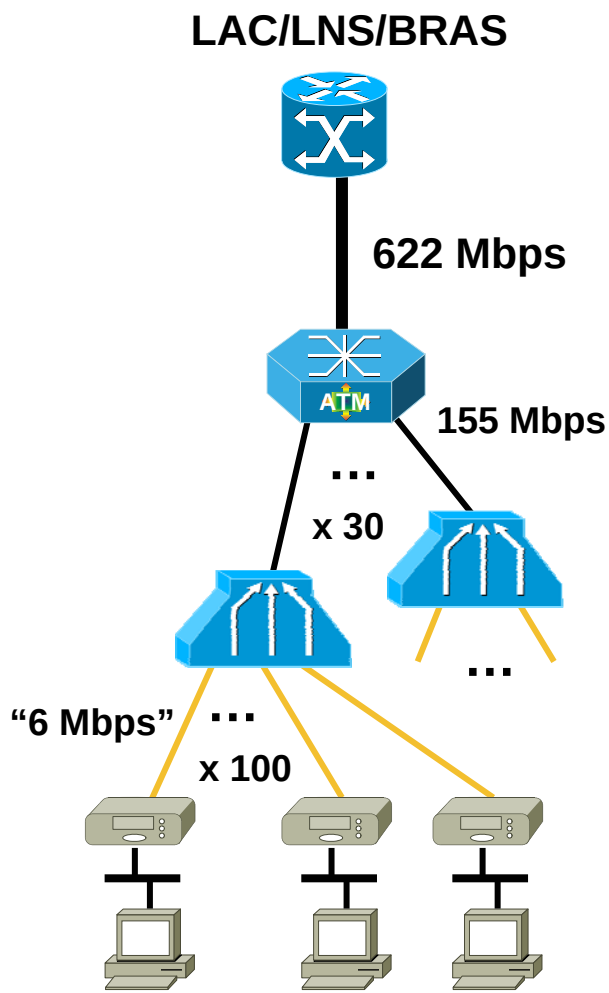
- **Unicast:**
  - **rfc1918 addresses**
  - **Reuse of host addresses**
  - **“privacy” for hosts**
- **Multicast:**
  - **IPv4: 239.0.0.0 / 8 addresses**
  - **IPv6: 16 scopes in architecture**
  - **Geographic form of access control for applications.**
  - **No per source/receiver control**
  - **Reuse of ASM group addresses**



Example scoped Address architecture with IPv4 multicast

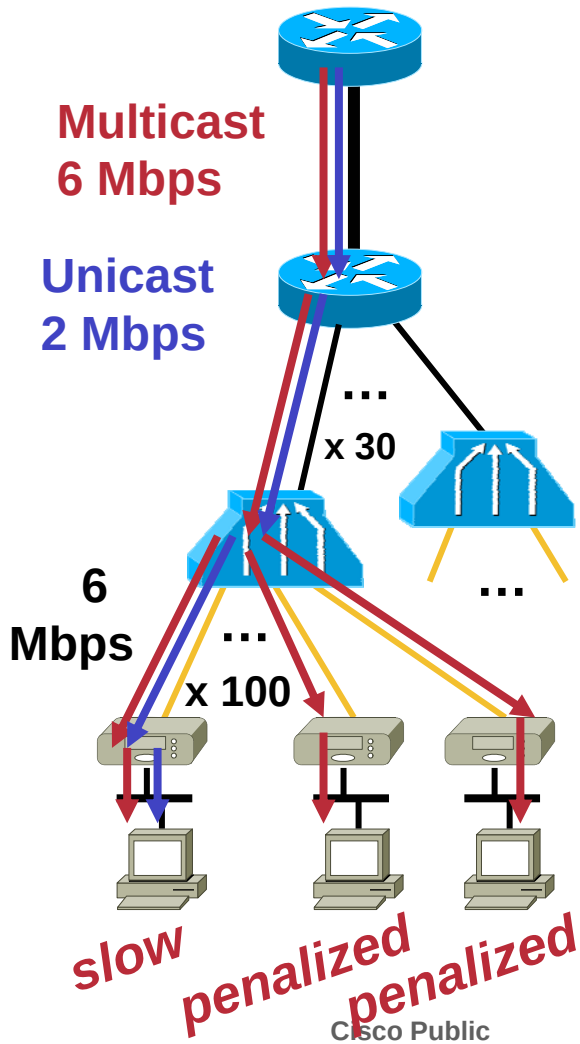
Multicast vs. Unicast
Application Side Difference- Unicast/TCP

- **"Admission" control – unicast**
  - **Relies on congestion control**
- **Up to 30 times *oversubscription***
- **Works!**
  - **Time-statistical multiplexing**
  - **95% TCP (or like TCP)!**
    - **Reliability by retransmission**
    - **Sender rate adoption**
      - **WRED, DPI, … in network**
- **Non-real-time / best-effort service**

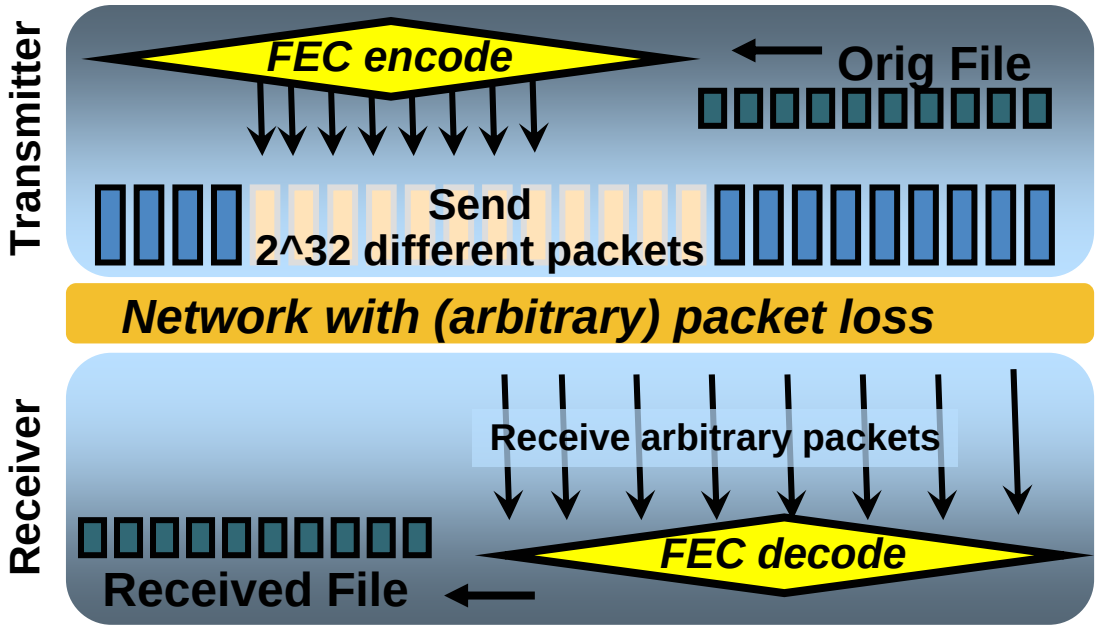


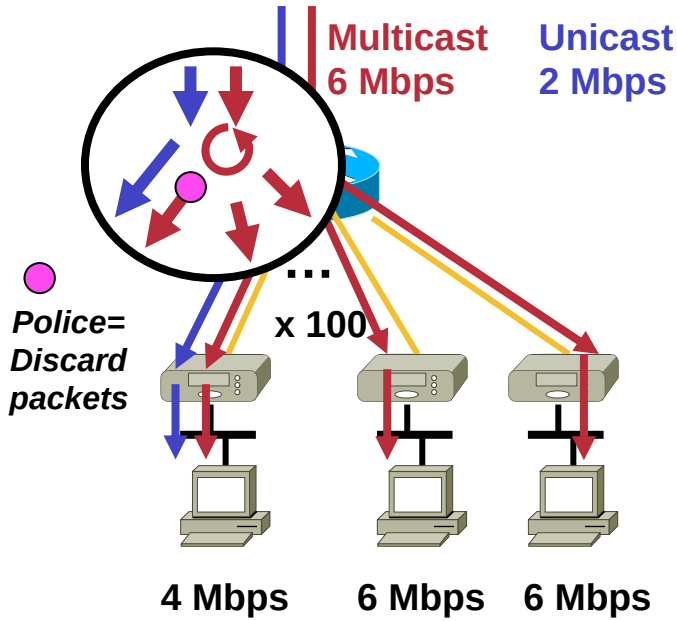
Multicast vs. Unicast
Application Side Difference – Multicast/PGM

- **PGM**
  - Multicast equivalent of TCP
  - FEC/NAK retransmission
  - PGM-CC (congestion control)
    - Match sender rate to slowest receiver. WRED/TCP compatible
- **Multicast problem**
  - Penalization by slowest receiver!
  - Fate sharing between receivers
    - … Ignore too slow receivers
    - Best with enterprise apps ?



Multicast vs. Unicast
Application Side Difference - Multicast/Large-Block-FEC





- ALC with large-block FEC (Tornado/Raptor/..) codec
- Sustain arbitrary packet loss and still decode content
- Just discard packets at every hop under congestion

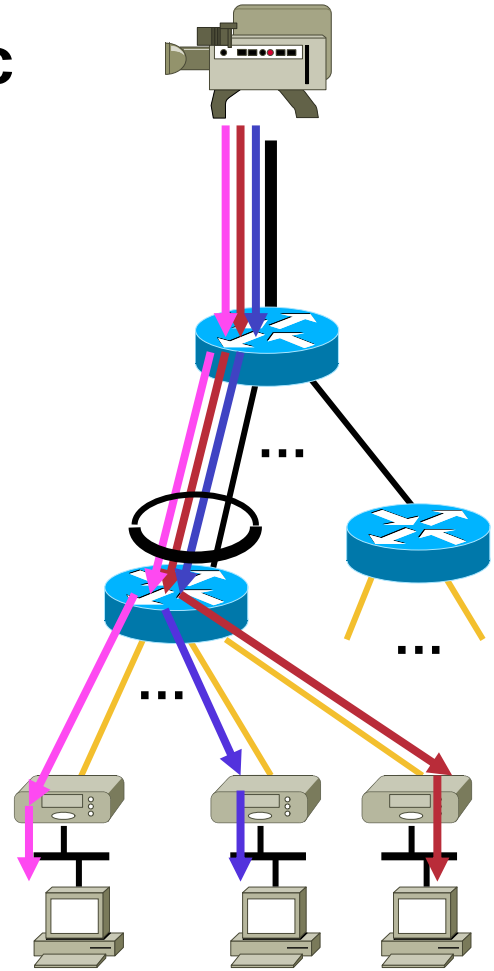
  Use even less-than-best-effort class (scavenger)

Multicast vs. Unicast
Application Side Difference- Real-Time Traffic

- **Congestion and real-time traffic**
  - **Small % of today's unicast traffic**
  - **Large % of today's multicast traffic !!!.**
- **Temporary "congestion"/BER caused packet loss**
  - **(short-block) FEC (MPEG)**
  - **retransmissions (TCP, PGM)**
- **Longer term "congestion"/oversubscription:**
  - **Over-provisioning / Diffserv bandwidth allocation.**
  - **Sender codec rate adaptation**
  - **Per-flow admission control (Intserv)**

Oversubscription with Real-Time Traffic

- **Consider link filled with real-time traffic**
  - **E.g.: 100Mbps link, 4Mbps TV**
- **Can fit up to 25 Flows**
- **What happens when 26th flow is put onto link ?**
  - **All 26 flows become useless**
- **Problem for unicast (VOD) and multicast (broadcast)**
  - **But solutions can differ**



Multicast vs. Unicast
Sender codec-rate Adoption

- **Unicast Audio/Video:**

  Sender (encoder/transcoder): Reduce bit rate / lower quality due to congestion.

  Used with "Internet AV" streaming.

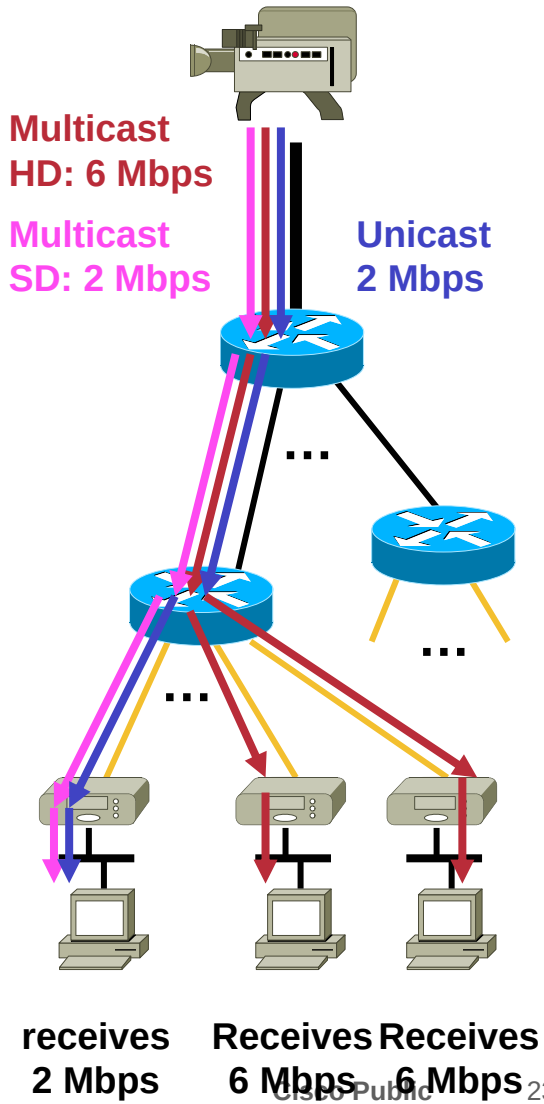
- **Multicast / "Simulcast" Audio/Video:**

  No third-party receiver penalizing:

  Send different BW encodings

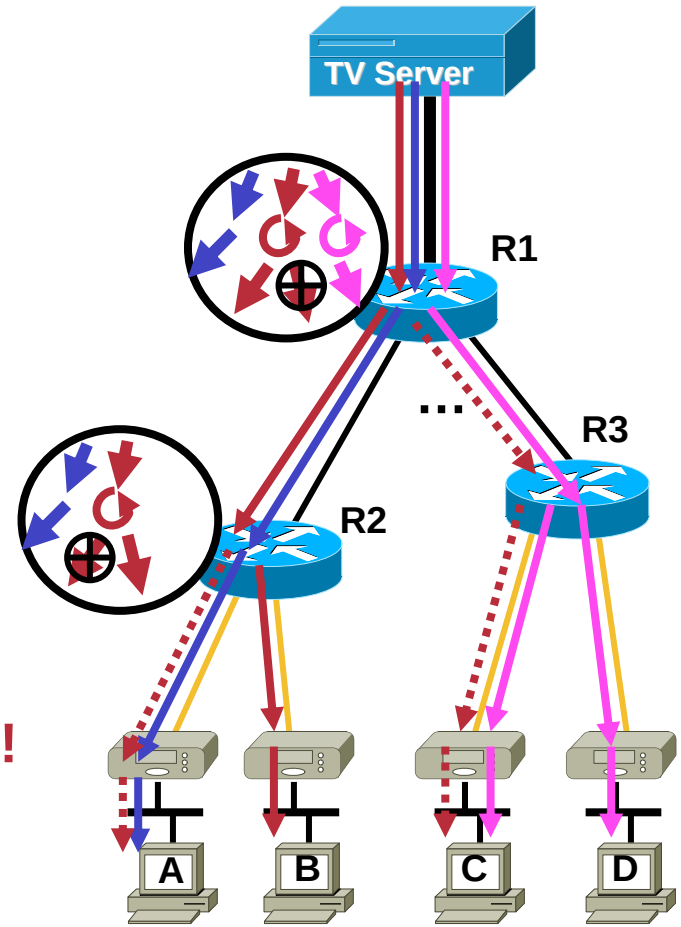
  Receivers choose BW/encoding by joining to specific group/channel.

  Less-granular than unicast



Multicast vs. Unicast
Application Side Difference – Intserv Admission Control

- **Intserv:**
  - per flow (admission) control
- **Unicast:**
  - Source side enforcement!
  - No need for network enforcement
- **Multicast:**
  - Network enforcement!
  - Block forwarding at replication points!
- **Mechanisms:**
  - RSVP (unicast), CLI (mcast)



Multicast vs. Unicast
Summary

- **Unicast:**

  MUST protect hosts AND network nodes against attackers sending traffic

- **Multicast**

  Protect routers/switches against too much state

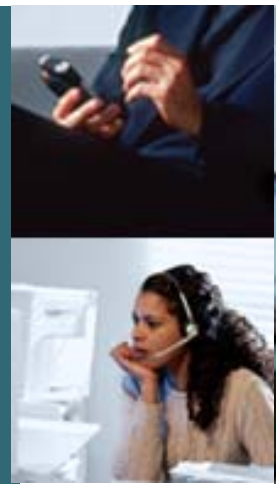
  ASM: MUST protect applications against unwanted sources.

    PIM-SM add. control plane protection (DR/RP)

  Per flow admission control

  Can control application participation and traffic flow due to explicit (*,G), (S,G) easier

Control Plane Security



Control Plane Security

- IGMP / MLD / PIM (RP/DR) / MSDP / AutoRP / SAP
- Spoof function (RP, DR, BSR, MA, MSDP-peer)
  - DoS network, participants
- Create administrative boundaries
- Overload router (control plane) CPU
- Create non-permitted state
  - Covered in access-control section
- Memory (SW) and hardware (state) overload
  - Covered in admission-control section

Filter for Control Plane Packets Non-Multicast Specific

- ip receive access-list <ext-acl>
- Filter applied to “received” packets
  - Unicast to router interface addresses
  - IP Broadcast
  - Packets with router alert option
  - Packets for joined IP multicast traffic
    - Link local scope groups (“show ip int”).
    - Groups with “L” flag set
- If not available – replace with per-interface ACL!

Filter for Control Plane Packets Non-Multicast Specific

- **Not incoming interface specific!**
  - **Can not filter per-subscriber or user vs. backbone interfaces**
- **Usage guideline / examples:**
  - **Positive list of required control plane packets**
    - **Protects against unknown stuff the router may accept without the admin knowing**
  - **Negative list of unwanted protocols. E.g.:**
    - **Unicast PIM packets (not RP / candidate-BSR)!**
    - **Unicast IGMP packets (non UDLR use)**
    - **TCP packets to port 639 (MSDP) from non-MSDP peer sources (if MSDP running)**

MQC – Modular QoS CLI
Policing (Rate-Limiting) and Shaping of Packets

- **MQC: IPv4/IPv6 infrastructure in IOS**
  - **not multicast specific – multi purpose**
- ***Police received multicast control plane packets***
  - ***Protection against DoS attacks by large amounts of control plane packets***
- **Police received data packets**
  - **Enforce SLA bandwidth (e.g.: video flows)**
  - **For aggregate multicast traffic or individual flows**
- **Not security related:**
  - **Queue outbound multicast data packets according to diffserv classes**
  - **Shape outbound multicast data packets to control burstyness**
  - **Tag packets with DSCP bits according to application->Diffserv policies**

MQC – Modular QoS CLI
Three Tier CLI Hierarchy

- **As required**
  - **Define ACLs for reqd. objects**

1. **Traffic classification**
   - **identify traffic and assign to classes**
2. **Define the Diffserv policy**
   - **Assign classes to a policy**
   - **Define the Diffserv treatment for each class**
3. **Attach the Diffserv policy to a logical/physical interface**
   - **The point of application of a QOS policy**

```
ip access-group extended mc-control-acl
  permit ip any 224.0.0.0 0.0.0.255

class-map match-all mc-control-class
  match access-group mc-control-acl

policy-map mc-control-policy
 class mc-control-class
    police rate 4000000 bps
       burst 2500000
     conform-action transmit
     exceed-action drop
!
interface Serial0
 ip address 192.168.2.2 255.255.255.0
 service-policy input mc-control-policy
```

- **Example: Rate-limit aggregate of all multicast control packet into router to 4 Mbps (~5000 packets/sec @ 100 byte packets) – 5 sec burst.**

MQC – Modular QoS CLI
Three Ways to Use MQC

- **Explicit service policies on interfaces**
  - **standard case**
- **Microflow-policing**
  - **Automatic creation of service policies for individual flows**
- **CoPP - Control Plane Policing**
  - **Apply to control plane packets (policing only)**
  - **Same rules as for ip receive acl**
    - **Control-Plane**
      - **Establishes context like “Interface”, but allows only MQC commands afterwards**

Interface Filtering and Rate-Limiting on Control Packets

```
ip access-list extended no-control-in
  deny   pim  any any              ! Protect CPU against unwanted protocols
  deny   ospf any any              ! Run on router !
  permit igmp any host 224.0.0.22 ! Only IGMPv3
  deny   igmp any any
  permit ip   any any

access-list 101 permit any host 255.255.255.255 ! E.g.: DHCP requests
access-list 101 permit any 224.0.0.0 0.0.0.255  ! E.g.: IGMP memberships
Access-list 101 permit any host 192.189.1.1     ! Interface addr of router

Interface ethernet 0                            ! User facing
ip access-group no-control-in in
  ip igmp version 3
  ip igmp limit 2                               ! Allow 2 STBs
  rate-limit input access-group control-packets 8000 1500 200
    conform-action transmit exceed-action drop
```

- **Full throttle paranoia: HW-filter unwanted control packets, and HW-rate limit required ones (e.g.: DHCP, IGMP). Limit IGMP state.**
  - **Per-interface filter because of backbone interfaces**
  - **Per-interface rate-limiting to isolate amongst multiple users**

IGMP/MLD Packets

- **IPv6: MLD uses ICMPv6 protocol type packets**
- **IPv4: IGMP is a protocol type:**

  PIMv1, IGMPv1,v2,v3, mrinfo, DVMRP, mtrace

  IOS: all these protocols enabled (if multicast is)

  Bad ?:

  PIMv1 – legacy protocol behavior

  Mrinfo - eavesdropping (use SNMP)

  DVMRP – flood & prune

  Good:

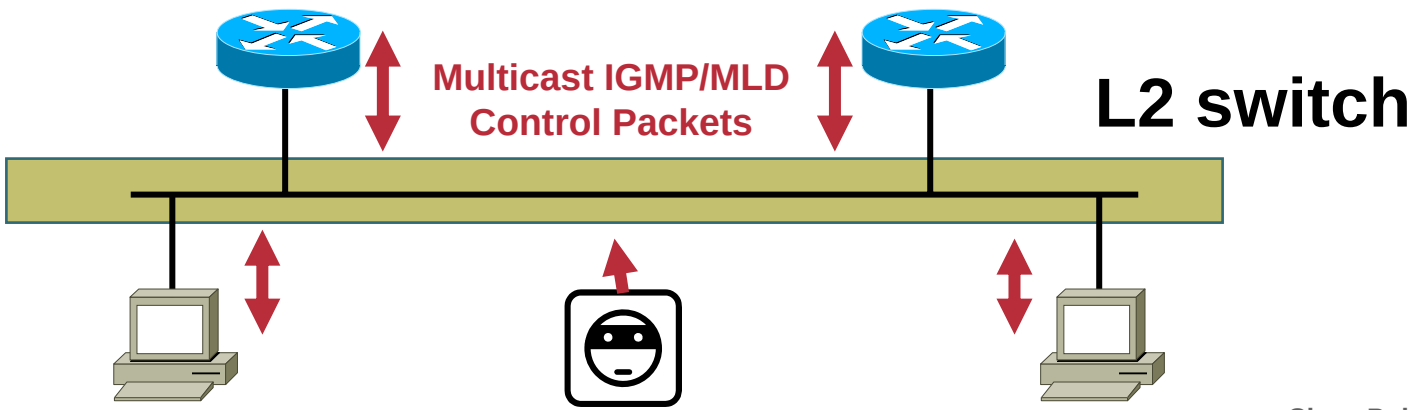
  mtrace – multicast equiv. Of traceroute

IGMP/MLD Protocol Packets

- **Bad: Unicast IGMP packets – for IGMP/UDLR**
- **Good: “Normal” = Multicast IGMP packets:**
  - **Attacks must originate on the same subnet**
  - **Link local multicast, not routed!**
    - **Memberships: Only with IGMPv3**
- **Forged query packets**
  - **Lower version: inhibit SSM, leave-latency**
  - **Bursts: response storms**
- **Forged (multicast) membership reports**
  - **Not a problem !?**

IGMP/MLD Protocol Packets
The L3 vs. L2 Problem

- **Forged queries:**
  - **Need inhibit that other hosts receive it!**
- **Forget membership reports**
  - **Forge IP address of other host**
  - **Router can not validate identity of hosts !**
- **L2 - per-port control required for these**
- **Per-LAN vs. per port access/admission control**



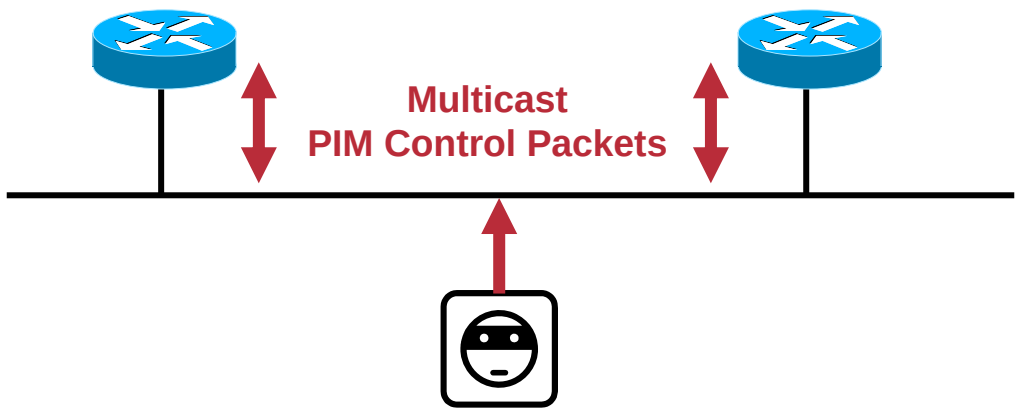
IGMP Packets
Example Extended ACL for Various IGMP Packets

```
ip access-list extended control-packets
  …
  deny   igmp any any  pim         ! No PIMv1
  deny   igmp any any  dvmrp       ! No DVMRP packets
  deny   igmp any any  host-query  ! Do not use with redundant routers !

  permit igmp any host 224.0.0.22 ! IGMPv3 membership reports
  permit igmp any any  14          ! Mtrace responses
  permit igmp any any  15          ! Mtrace queries
  permit igmp any 224.0.0.0 15.255.255.255 host-query  ! IVMPv1/v2/v3 queries
  permit igmp any 224.0.0.0 15.255.255.255 host-report ! IGMPv1/v2 reports
  permit igmp any 224.0.0.0 15.255.255.255 7           ! IGMPv2 leave messages
  deny   igmp any any              ! Implicitly deny unicast IGMP here!
  …
  permit ip   any any              ! Likely deny any on control plane!

ip receive access-list control-packets

interface ethernet 0
  ip access-group control-packets in          ! Could put filter here too
```

- ***http://www.iana.org/assignments/igmp-type-numbers***
- ***Numeric 'port' <n> = IGMP type number 0x1<n>***

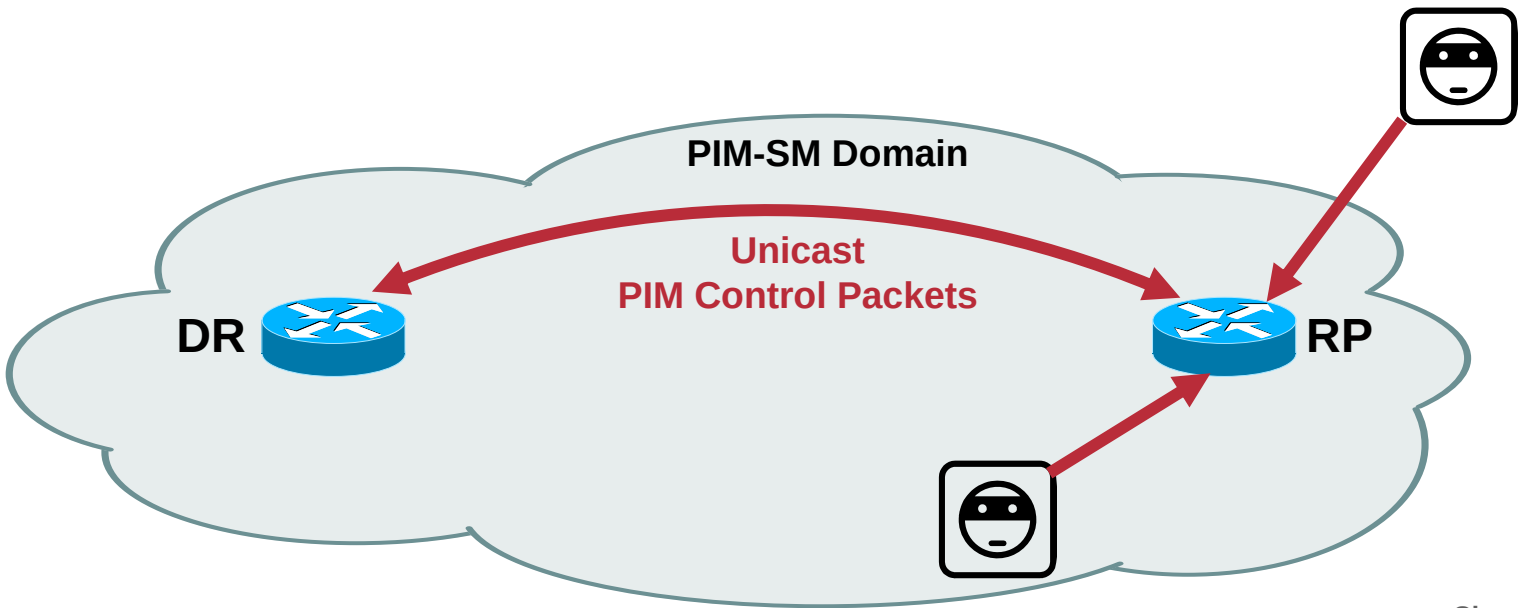
PIM Packets – Multicast

- **Multicast PIM Control Packets :**
  - **Hello, Join/Prune, Assert, Bootstrap, DF-elect**
  - **All are link local multicast (TTL=1)**
  - **All are multicast to All-PIM-Routers (224.0.0.13)**
- **Attacks must originate on the same subnet**
  - **Forged Join/Prune, Hello, Assert packet.**



PIM Packets – Unicast

- **Unicast PIM Control Packets :**
  - **Register: Unicast from DR to RP.**
  - **Register-Stop: Unicast from RP to DR.**
  - **C-RP-Advertisement: Unicast from C-RP to BSR.**
- **Attacks can originate from anywhere!**



PIM Packets – Auto-RP

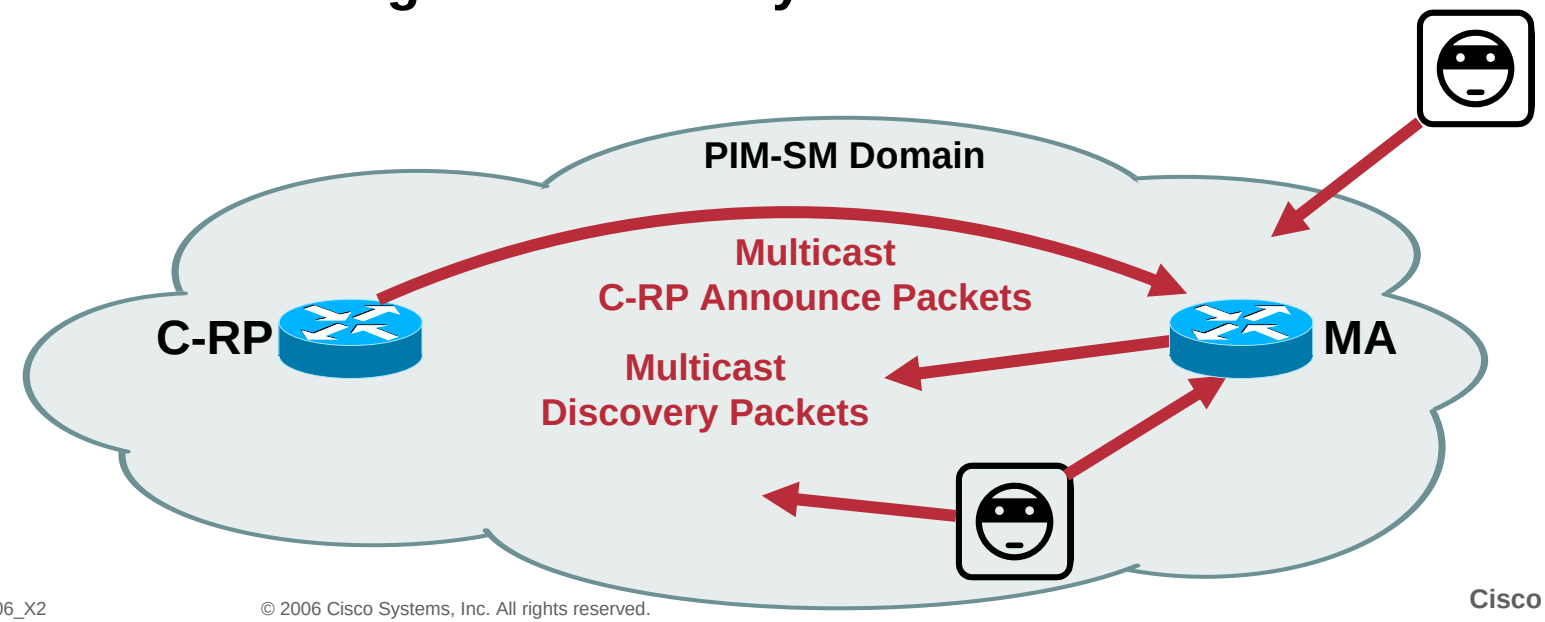
- **IOS: AutoRP/BSR always enabled, non-configurable**
- **Auto-RP PIM Control Packets :**

  C-RP-Announce: Multicast (224.0.1.39) to all MA's.

  Discovery: Multicast (224.0.1.40) to all Routers.

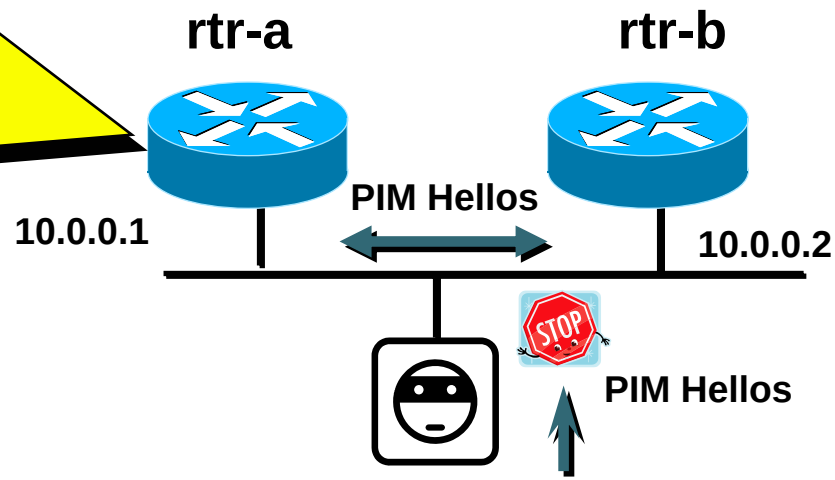
  Normally Dense mode flooded!

- **Attacks can originate from anywhere!**



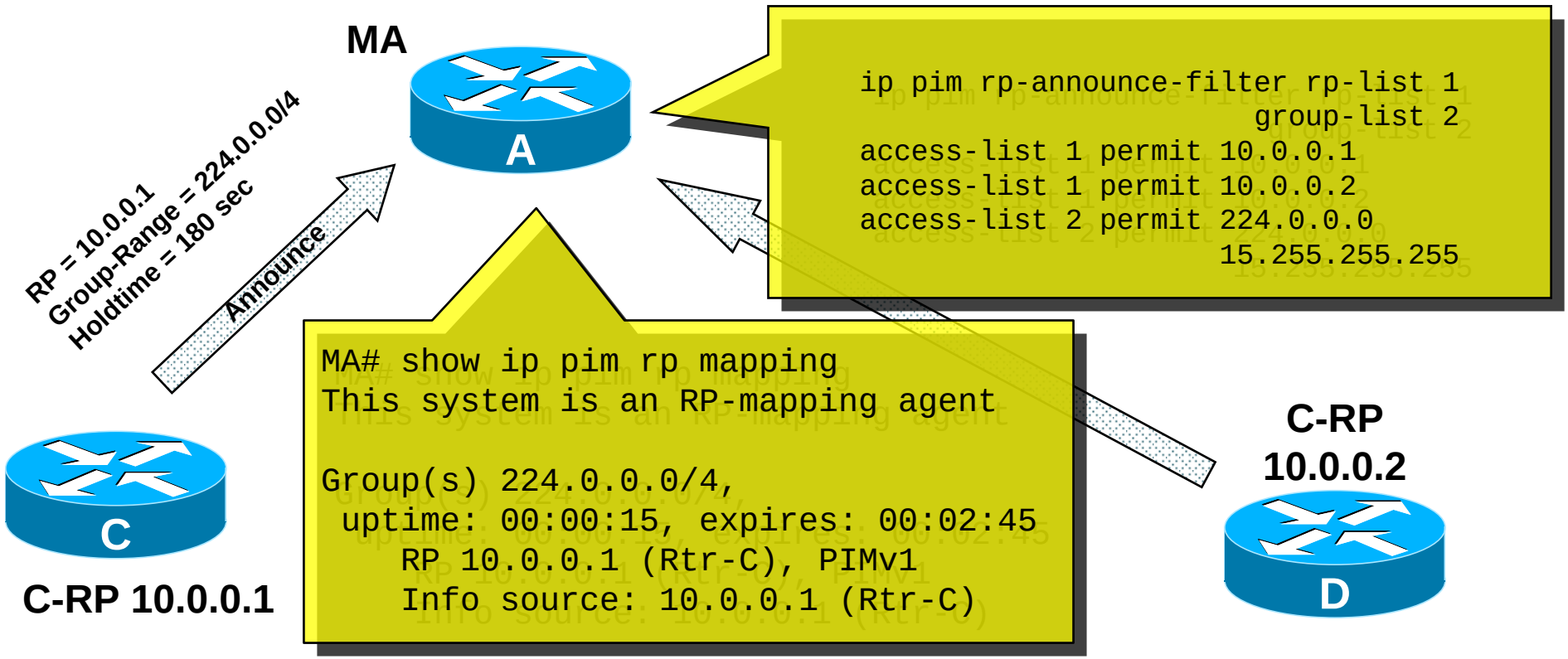
PIM Neighbor Control

```
ip multicast-routing
ip pim sparse-mode
ip multicast-routing

access-list 1 permit 10.0.0.2
Access-list 1 deny any

Interface e0
  ip pim sparse-mode
  ip pim neighbor-filter 1
```



- **Must receive Hellos to establish PIM neighbor**
  - **DR election, failover**
  - **Accept / Send PIM Join/Prune/Assert**
- **Use ip pim neighbor filter to inhibit neighbors**
  - **Filters all PIM packets from non-allowed sources**
  - **Hellos, J/P, BSR, … !**

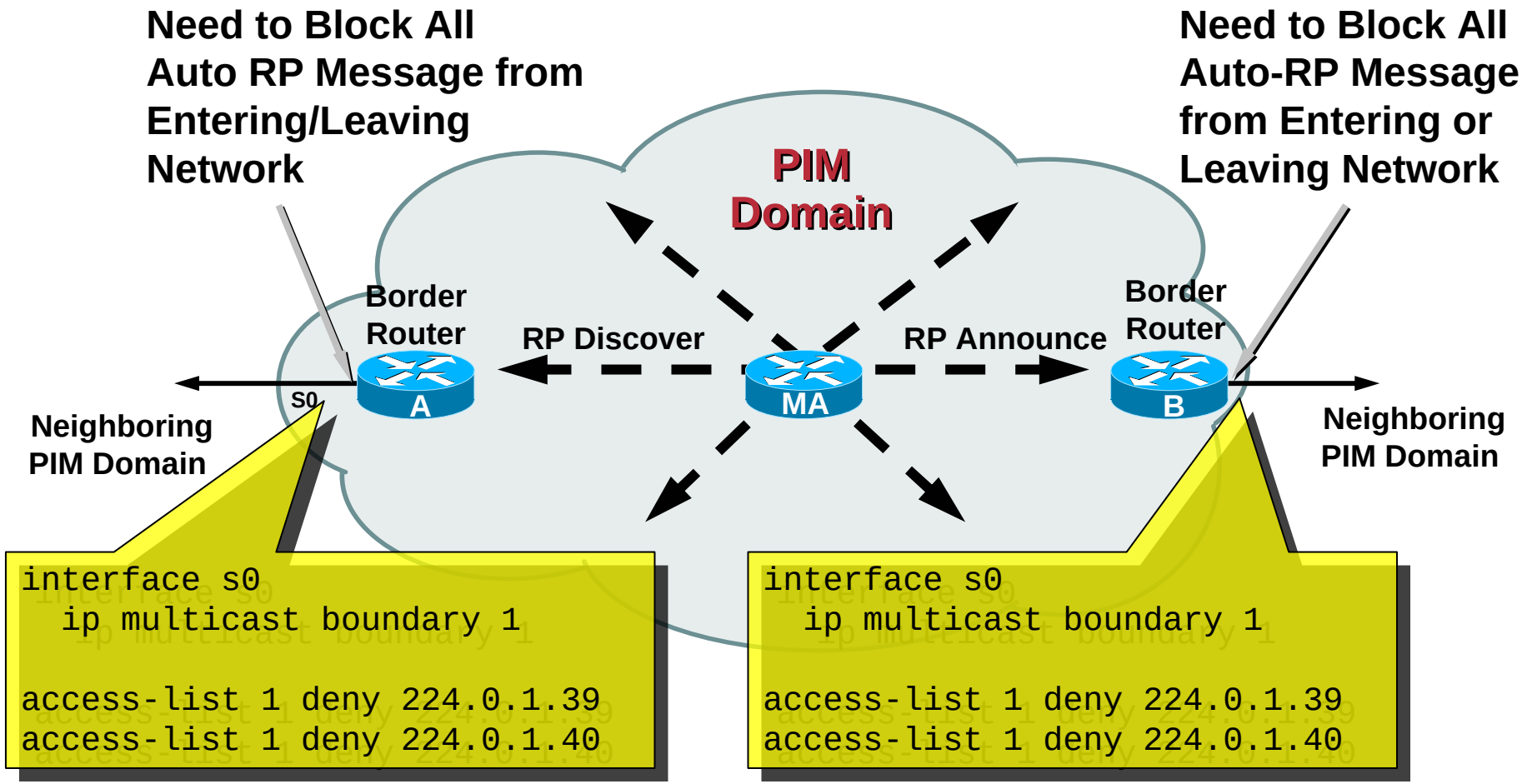
AutoRP Control
RP Announce Filter



- **ip pim rp-announce-filter**

  Configure on MA which router (IP-addr) is accepted as C-RP for which group ranges / group-mode

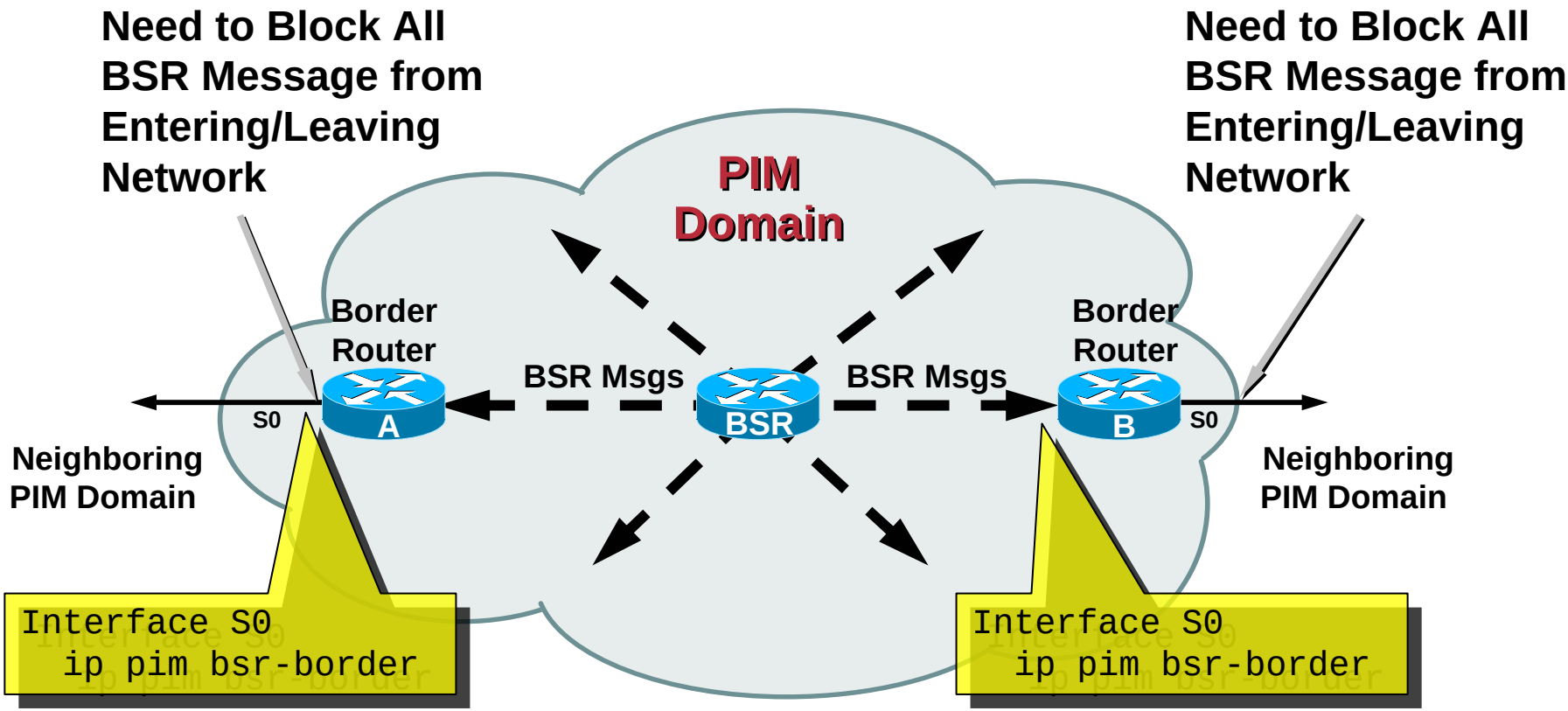
Auto-RP Control
Constrain Auto-RP Messages



- **AutoRP packets:**

  224.0.1.39 (RP-announce), 224.0.1.40 (RP-discover)

BSR Control
Constrain BSR Messages



- **ip pim bsr-border**

  Filters messages (multicast) from BSR

  no ACL possible (hop by hop forwarded)

Protection Against Attacks on Access Networks

- **Attacks by hosts (multicast)**
  - **PIM Hellos – become DR – no traffic forwarded to LAN**
  - **PIM joins – receive traffic (should use IGMP / filtered)**
  - **AutoRP RP-discovery or BSR bootstrap**
    - **Announce fake RP, bring down SM/Bidir service**
- **Attacks by hosts (unicast)**
  - **Send register/register-stop**
    - **Inject fake traffic**
    - **BSR announce packets – announce fake RP**
- **Hosts should never do PIM !**

Protection Against Attacks on Access Networks

- ip pim multicast boundary - filter AutoRP (not PIM) –
- ip pim rp-address override - not using AutoRP / BSR
- **Non-redundant access-network**

  ip pim announce-filter acl-none
  Multicast only: need to add filtering on RP/BSR

  ip access-group acl-deny-all-pim in

- **Redundant access-networks**

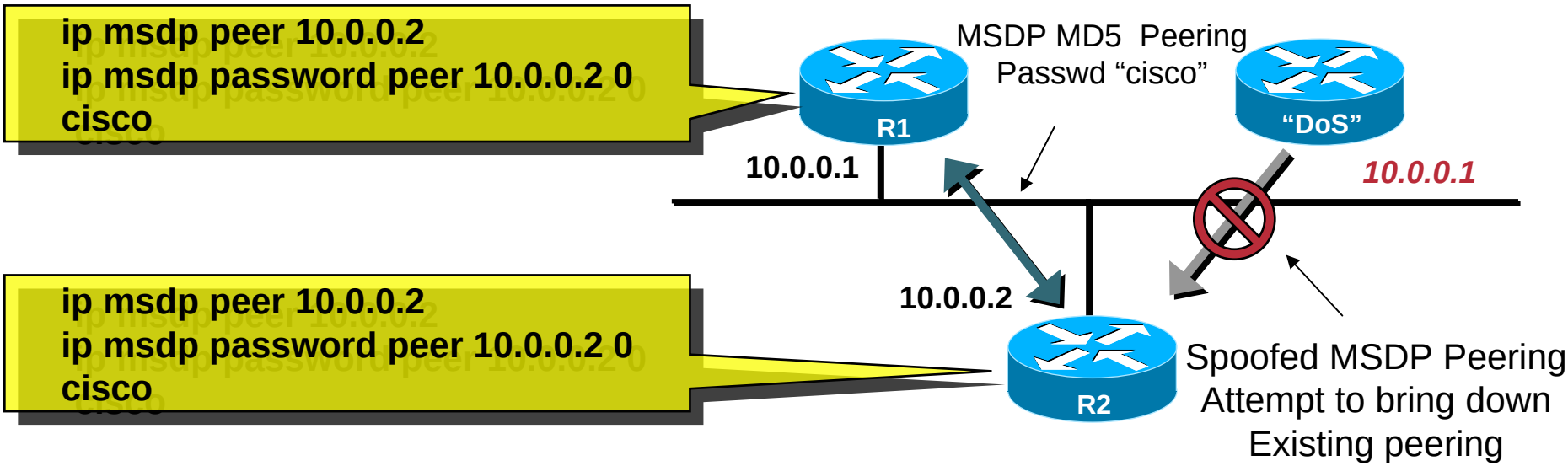
  Allow PIM from redundant neighbor.
  Need L2 mechanisms to inhibit forged packets !

  Add ip pim bsr-border - group-mapping attacks

PIM Control Plane
Missing Simplifications / Improvements

- **Disable AutoRP operations on router**
- **Disable BSR operations on router**
- **Disable RP operations on router**
  - **Functions not running can not be attacked**
- **And/Or:**
- **Authentication for PIM / AutoRP messages**
  - **Differentiate PIM / AutoRP messages from forged**
    - **Without knowledge of IP addresses of legal peers**
  - **Possible with IPsec for multicast (later slides)**
    - **Not easy to configure yet though**

MSDP MD5 Password Authentication



- **Protect MSDP peering against spoofed packets**
  - **Protects against spoofed sourced packets**
  - **Partial protect against man-in-middle**
- **Uses RFC2385 TCP authentication header**
  - **Defined for BGP**
  - **Actually independent of BGP**

SAP / SDP / SDR

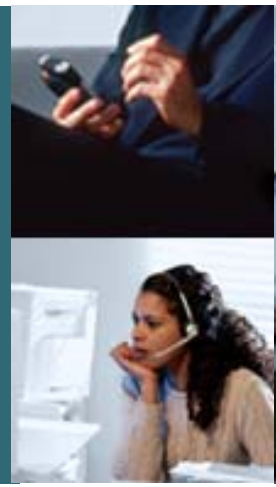
- ip sap listen
- Receive SAP/SDP messages
  - "Just" for show ip sap output
  - Not used / required by router otherwise
    - *except legacy ip multicast rate-limit function*
- Dos against router CPU / memory
- Recommendation
  - Do no enable!
    - *unless considered important to troubleshoot*
  - If enabled, use CoPP to rate-limit

Other Control Plane Security Features

- **ip multicast mrinfo-filter <std-acl>**

  Limit mrinfo answers to specific requesters

Access Control
Includes Scoping

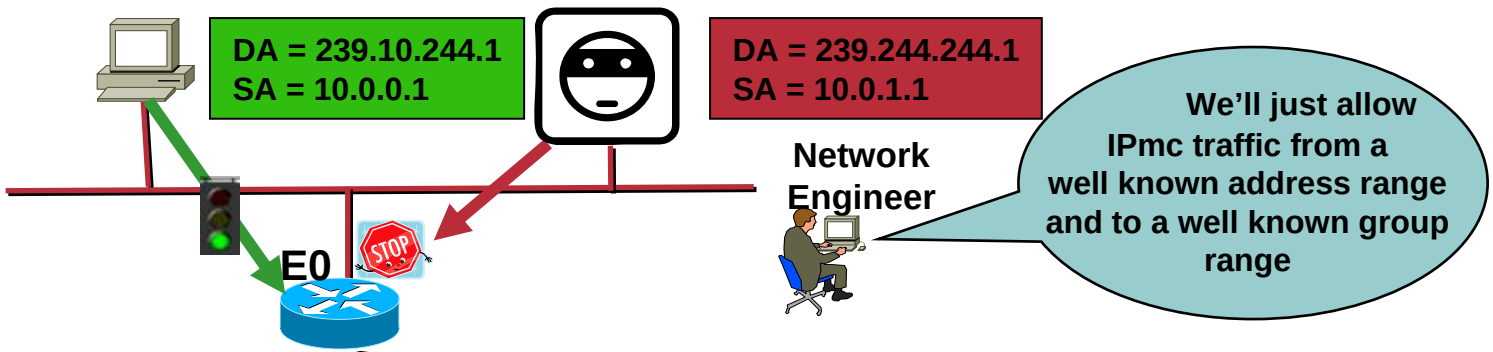


Access Control Overview

- ***Control which ASM groups and SSM channels systems (hosts or subscribers) can send and/or receive traffic for***
- **ip access-group / ipv6 traffic-filter**
- **ip pim accept-register / ipv6 pim pim accept-register**
- **ip igmp access-group / ipv6 mld access-group**
- **Ip multicast group-range / ipv6 multicast group-range**
- **ip multicast boundary / ipv6 multicast boundary**

Packet Filter Based Access Control

ip access-group [in|out] / ipv6 traffic-filter [in|out]



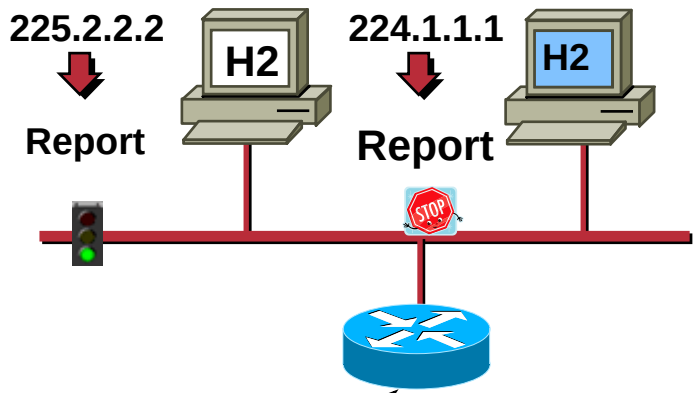

```
ip access-list extended source
  permit ip host 10.0.0.0 0.255.255.255 239.0.0.0 0.127.255.255
  deny   ip any                 224.0.0.0 15.255.255.255 log
  deny   ip any                 any log

interface ethernet0
  ip address 10.1.1.1 255.255.255.0
  ip access-group source in
```

- **HW installed on most platforms - (costs HW filter)**
- **Filters before multicast routing – no state creation**
- **Best for ingress – egress filtering best at multicast routing**

Host Receiver Side Access Control

ip igmp access-group / ipv6 mld access-group

- **Filter group/channels in IGMP/MLD membership reports**
  - **Controls entries into IGMP/MLD cache**
  - **Extended ACL semantics like multicast boundary**
  - **Deny only effective if protocol = ip**
  - **IGMPv2/MLDv1 reports:**
    - **source = 0.0.0.0 / 0::0**

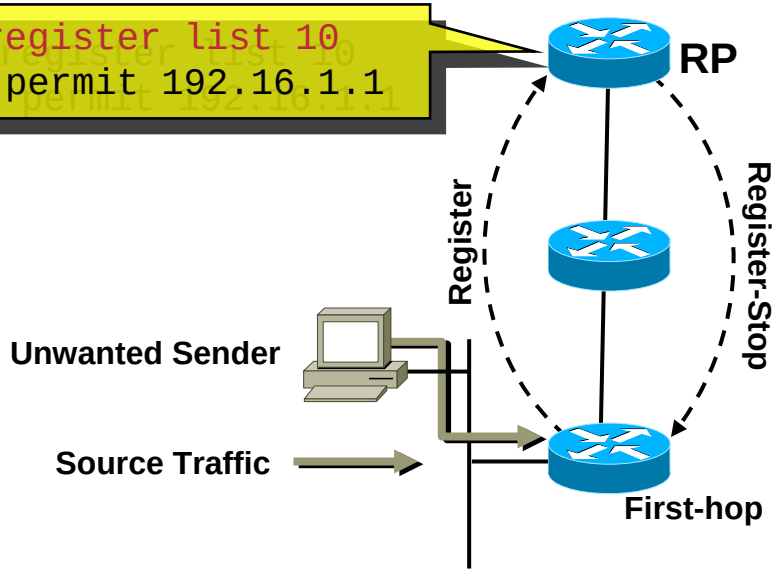



```
ip access-list extended allowed-multicast
  permit ip any host 225.2.2.2       ! Like simple ACL
  permit ip 10.0.0.0 0.255.255.255 232.0.0.0 0.255.255.255
  deny   ip any any

interface ethernet 0
 ip igmp access-group allowed-multicast
```

PIM-SM Source Control

ip pim accept-register / ipv6 pim accept-register

```
ip pim accept-register list 10
access-list 10 permit 192.16.1.1
```



- Unwanted source traffic hits first-hop router
- First-hop router creates (S,G) state and sends Register
- RP rejects Register, sends back a Register-Stop.

- RP-based (central) access control for (S,G) in PIM-SM
- Extended-Acl: which source can send to which group
- Imperfect:
  - (S,G) state on FHR still created
  - (S,G) traffic still to local and downstream rcvrs.

Disabling Multicast Groups

- **ip multicast group-range <std-acl> (future)**
- **ipv6 multicast group-range <std-acl> (12.4T)**
- **Disable all operations for groups denied by <acl>**
  - **Drop / ignore group in all control packets.**
    - **PIM, IGMP, MLD, MSDP.**
  - **No IGMP/MLD (cache), PIM, MRIB/MFIB state.**
  - **Drop all data packets.**
    - **HW-discarding platform dependent**

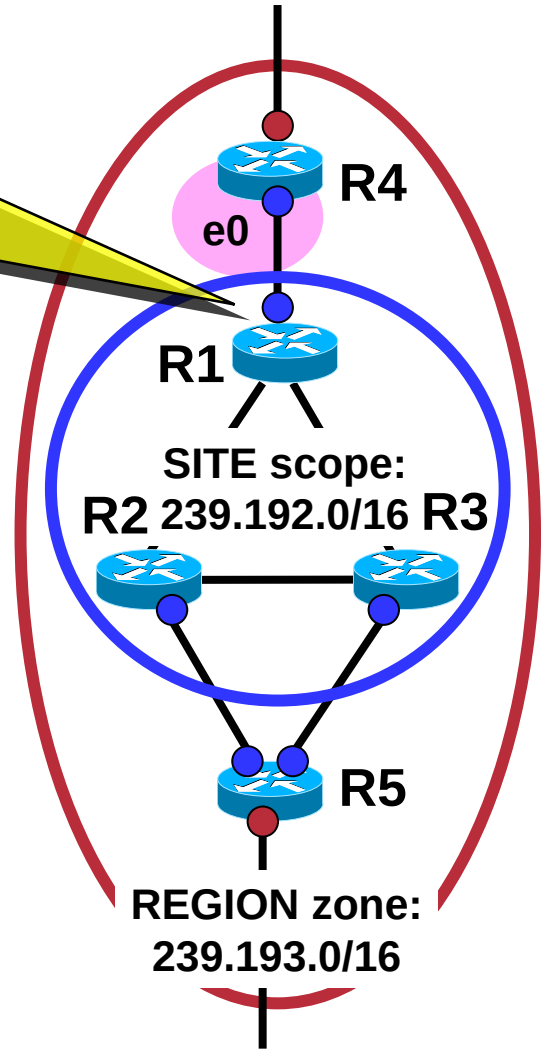
Interface / Protocol Level Access Control
Overview

- **IPv4: per-interface either or all of A, B ,C (one each)**
  - **A: ip multicast boundary <std-acl> [filter-autorp]**
    - **Group scope boundaries**
    - **Semantic filtering of AutoRP messages**
  - **B: ip multicast boundary <ext-acl> in**
  - **C: ip multicast-boundary <ext-acl> out**
    - **Extended form for access-control, SSM scopes**
- **IPv6: per-interface one config of A**
  - **A: ipv6 multicast boundary scope <n>**
    - **Scoping simple due to IPv6 architecture**
    - **No B, C options (yet)**

Interface / Protocol Level Access Control IPv4 Scope Boundaries

```
access-list 10 deny 239.192.0.0 0.0.255.255
access-list 10 permit any

Interface ethernet 0
 ip multicast boundary 10 [ filter-autorp ]
```

- **Configure on interfaces passing the (imaginary) scope boundary line**
- **Protocol filtering for denied groups**
  - **IGMP / PIM**
- **On inside and outside routers**
- **Use filter-autorp option when using AutoRP**
  - **No equivalent for BSR (yet)**



Interface / Protocol Level Access Control IPv6 Scope Boundaries

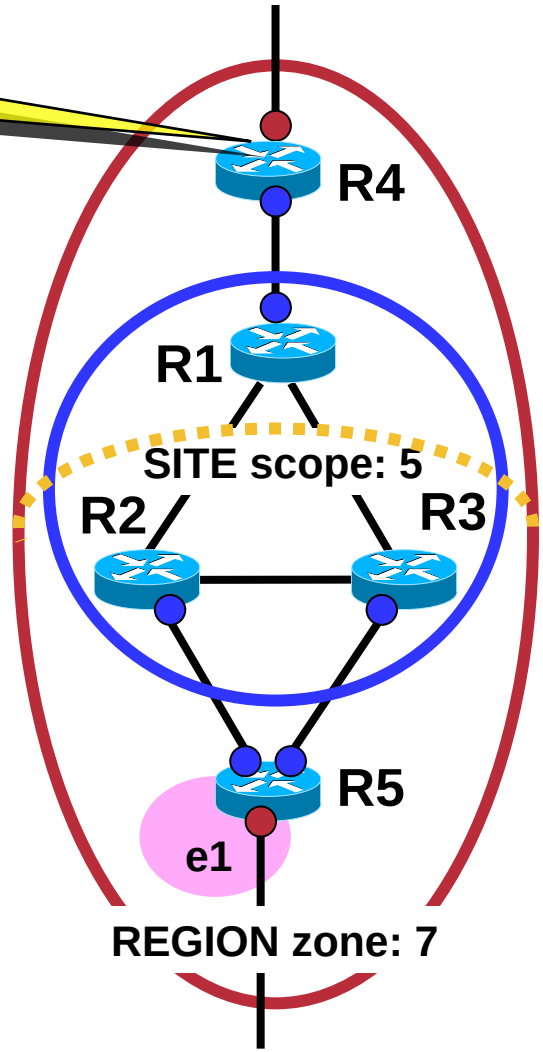
```
Interface ethernet 1
 ipv6 multicast boundary scope 7
```

- **Scope addresses fixed by architecture**
- **Boundary for scope <n> always also filters scope 2..<n-1> addresses.**

  Larger scope may not cut through smaller scope.

- **ACL for scope implicitly defined:**

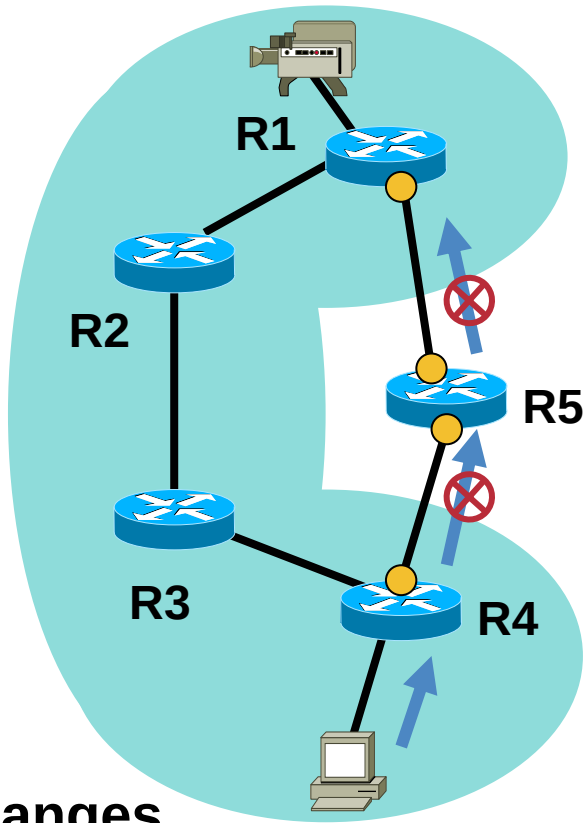
```
ipv6 access-group standard scope7filter
  permit ff02::00 00f0::00 ! Always permitted
  deny   ff03::00 00f0::00 ! Deny up to and
  …                        ! Including scope 7
  deny   ff07::00 00f0::00 !
  permit ff08::00 00f0::00 ! Permit higher
  …                        ! scopes
  permit ff0f:000 00f0::00 !
```



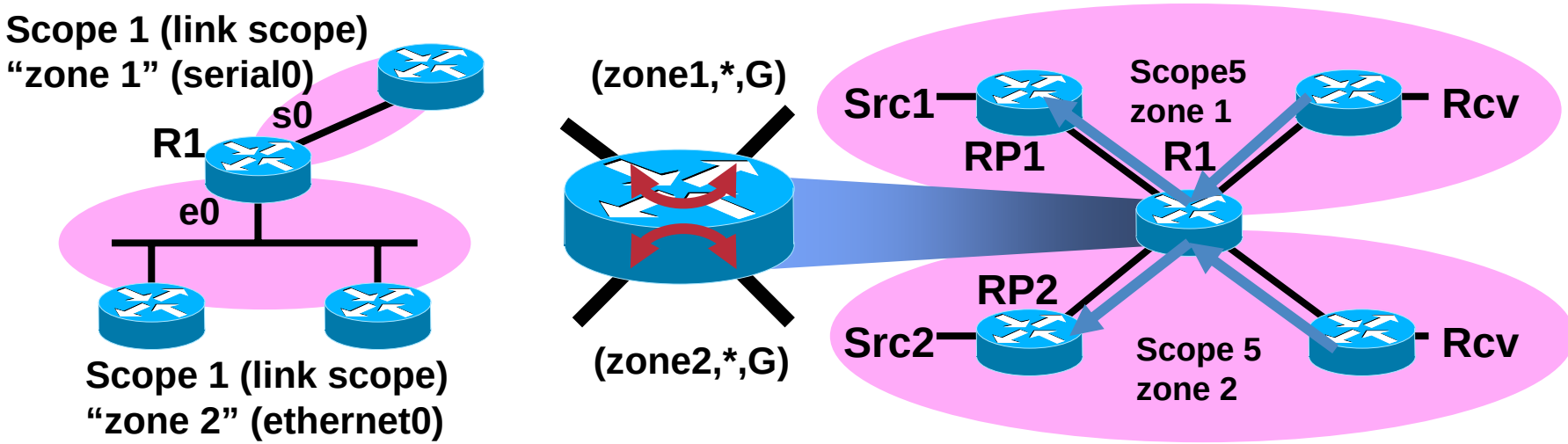
Interface / Protocol Level Access Control Shape and Structure of Scopes

- **Scopes must be convex.**
  - Traffic between source/receivers must not cross scope boundary.
  - Property of topology AND metric
- **Scopes contained in larger scopes?**
  - Only mandatory in IPv6 arch
  - Scope expanding ring search
  - Smaller scopes subset of larger scopes (IPv4, historical)
- **Recommendations IPv4**
  - Define IPv4 scopes as non-overlapping ranges.

Picture:
Non-convex scope



Interface / Protocol Level Access Control Location of Scope Boundaries



SUPPORTED *NOT SUPPORTED*

- **IPv6 arch: scope boundaries cut through router !**

  draft-ietf-ipngwg-addr-arch-v3-11.txt

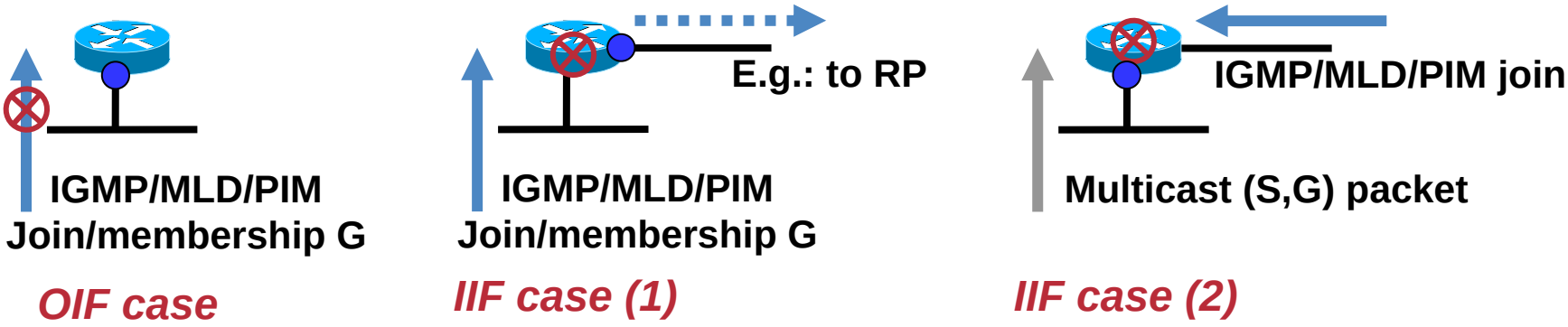
- **IOS / IOS-XR:**

  ALWAYS for link-local scope

  NEVER for larger scopes

  Scope boundaries only cut through links!

Interface / Protocol Level Access Control
Filtering Rules (1)



- ***Filter IGMP/PIM messages and create filtered state***
- **OIF (“out”) case:**

  Discard IGMP/PIM “join” for egress denied (*,G)/(S,G) state. Do not create state.

- **IIF (“in”) cases:**

  (1) Force OIF of mroute state to NULL if state denied on RPF-interface. No joins sent on RPF-interface

  (2) Directly connected traffic: discarded state with NULL OF list. Also inhibits PIM-SM registering.

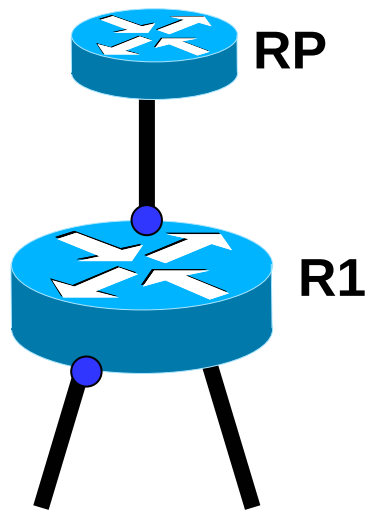
Interface / Protocol Level Access Control
Filtering Rules (2)

- PIM-DM: Do not flood on egress boundary. Send prune on ingress boundary
- **PIM-SM/SSM/DM: State creation by directly connected sources.**

  Not Catalyst-6500/Cisco-7600 (IPv4). Boundary-acl also filters packets.
- **Bidir-PIM: May not inhibit upstream traffic with multicast boundary**

  For Bidir-PIM groups, use “ip multicast boundary” AND “ip access group”.

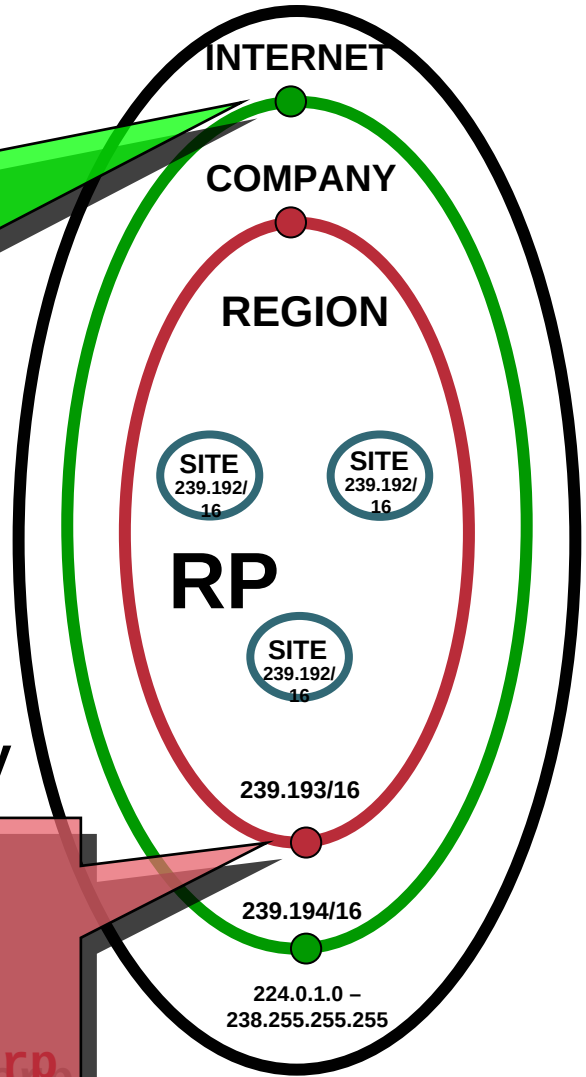
Picture: Boundaries and Bidir problem



Interface / Protocol Level Access Control AutoRP Filtering

- **Domain Boundary**

```
Access-list standard internet-boundary
  deny host 224.0.1.39
  deny host 224.0.1.40
  deny 239.0.0.0 0.255.255.255

Interface ethernet 0
  ip multicast boundary internet-boundary
```



Scope boundary – use filter-autorp

- **Group ranges intersecting denied ranges in ACL are removed from RP-Discovery/Announce messages at boundary**

```
Access-list standard region
  deny 239.193.0.0 0.255.255.255

Interface ethernet 0
  ip multicast boundary region filter-autorp
```

Interface / Protocol Level Access Control Separate In/Out Filtering

- *ip multicast boundary <ext-acl> in*
  - **IIF only - To inhibit traffic received on the interface**
- *ip multicast boundary <ext-acl> out*
  - **OIF only – Inhibit replication to the interface**
- **Extended ACL**
  - **Can filter each (*,G) and (S,G) state differently**
  - **Support SSM per-channel filtering.**
- **Inbound traffic must be permitted by both**
  ip multicast boundary acl and
  ip multicast boundary ext-acl in
- **Outbound traffic must be permitted by both**
  ip multicast boundary acl and
  ip multicast boundary ext-acl out

Interface / Protocol Level Access Control Semantics of Extended ACLs

```
Ip access-list extended ext-acl
  deny udp any 239.0.0.0 0.255.255.255  ! [1]
  deny pim any host 224.0.0.13          ! [2]
  deny ip  any host 224.0.0.13          ! [3]
  deny ip  host 0.0.0.0 any             ! [4]
  deny ip  any 239.0.0.0 0.255.255.255  ! [5]

Interface ethernet 0
  ip multicast boundary ext-acl out
  ip access-group       ext-acl out
```

- **ineffective for ip multicast boundary**

  [1], [2] Deny only effective with protocol “ip” (all packets of a (S,G)/(*,G)

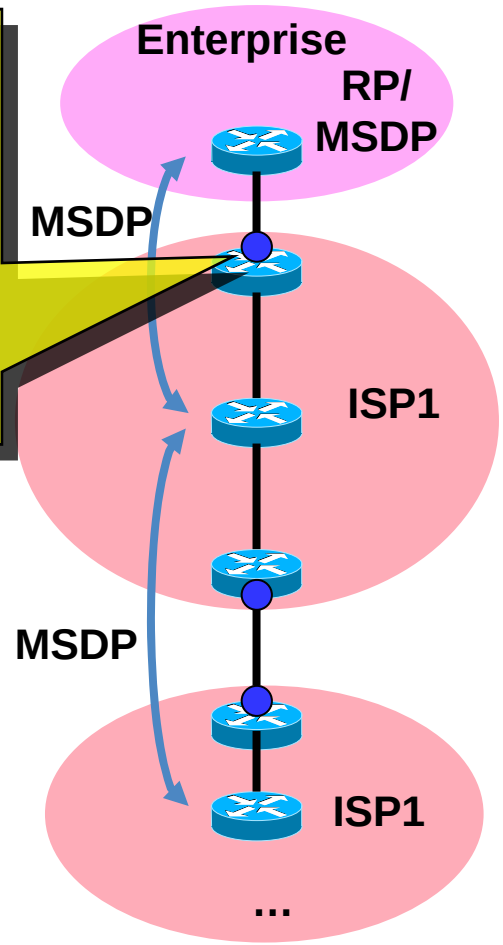
  [3] Can filter only routable groups!

- **Reuse ACL with ip access-group**
- **[4] Src = 0.0.0.0 = ***

  deny (*,G) joins / IGMPv2 memberships, but permit (S,G)

Interface / Protocol Level Access Control Example: interdomain (*,G) Filter

```
ip access-list extended interdomain-sm-edge
  deny   ip  host 0.0.0.0 any
  deny   ip  any          239.0.0.0 0.255.255.255
  permit ip  any          any

Interface ethernet 0
  ip multicast boundary interdomain-sm-edge out
  ip multicast boundary interdomain-sm-edge in
```



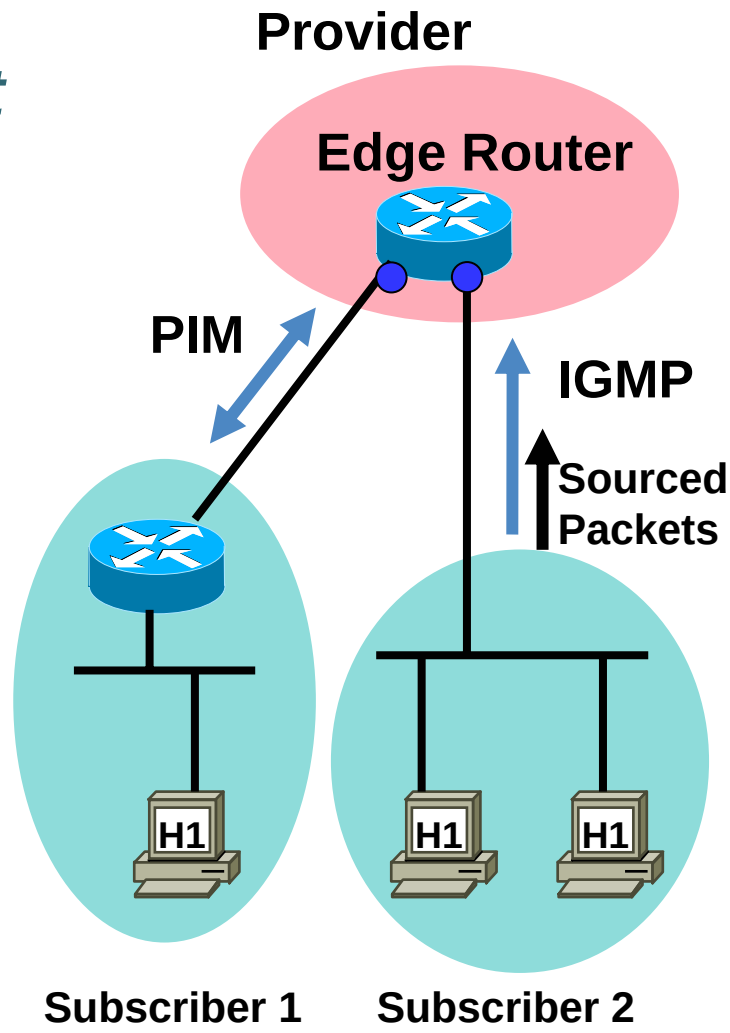
- **Example: PIM-SM transit provider:**
  - **MSDP peering only**
  - **(S,G), but no (*,G) on border**
  - **Inhibit (*,G) joins !**
  - **Also inhibit admin-scope addrs.**

Interface / Protocol Level Access Control Example: Subscriber Interfaces

- *ip multicast boundary ext-acl out* **supersedes** *ip igmp access-group ext-acl*
- **Use ip multicast boundary in/out independent of host or router subscriber**
- **Consider** *ip access-group ext-acl in*
- **Rule of thumb:**

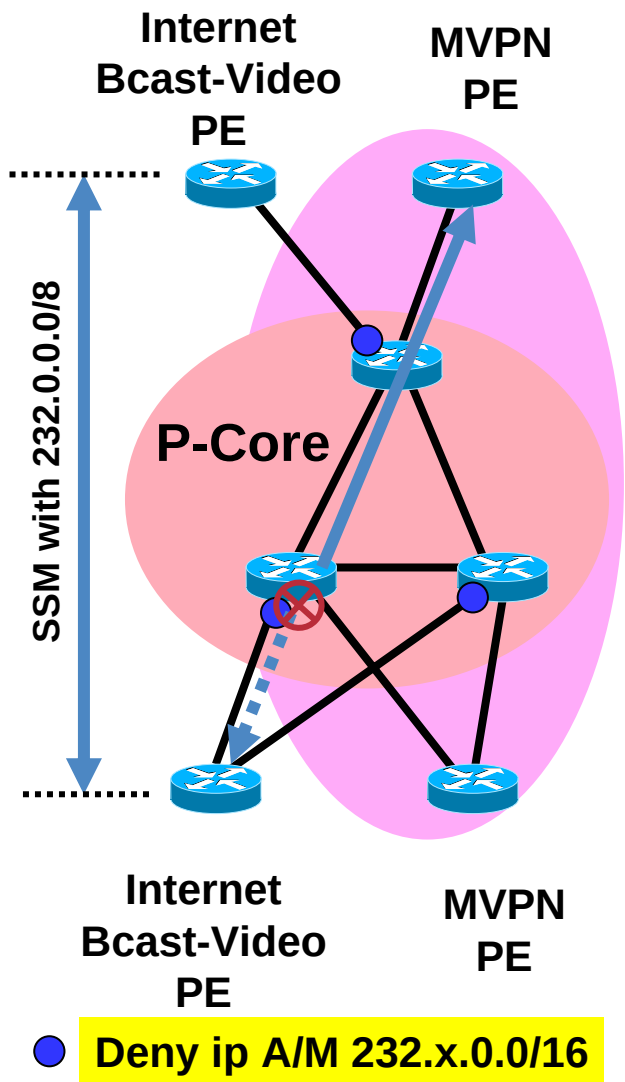
  Output: State based control

  Input: State {+ packet} control



Interface / Protocol Level Access Control
Example: Alternative SSM Scopes [(S/M, G/M)]

- **IPv6: 16 scopes for both ASM and SSM!**
- **IPv4: Only global scope SSM (232.0.0.0/8).**
- **Cisco recommendation, add ranges:**
  - **239.232.0.0/16 – admin scoped SSM**
  - **Not supported by all vendors**
- **SSM (S/M, G/M) scopes:**
  - **A/M = loopback of MVPN-PE**
  - **232.x.0.0 = MVPN Default & Data-MDTs**
  - **Use (S/M,G/M) scope filter on P links facing Internet-PE (non-MVPN)**
  - **Result:**
    - **Full Internet SSM transit**
    - **Protected MVPN service**



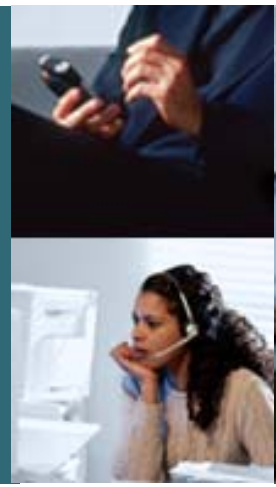
Access Control

Recommended Interdomain MSDP SA Filter

```
! domain-local applications
access-list 111 deny   ip any host 224.0.2.2     !
access-list 111 deny   ip any host 224.0.1.3     ! Rwhod
access-list 111 deny   ip any host 224.0.1.24    ! Microsoft-ds
access-list 111 deny   ip any host 224.0.1.22    ! SVRLOC
access-list 111 deny   ip any host 224.0.1.2     ! SGI-Dogfight
access-list 111 deny   ip any host 224.0.1.35    ! SVRLOC-DA
access-list 111 deny   ip any host 224.0.1.60    ! hp-device-
disc
!-- auto-rp groups
access-list 111 deny   ip any host 224.0.1.39
access-list 111 deny   ip any host 224.0.1.40
!-- scoped groups
access-list 111 deny   ip any 239.0.0.0 0.255.255.255
!-- loopback, private addresses (RFC 1918)
access-list 111 deny   ip 10.0.0.0 0.255.255.255 any
access-list 111 deny   ip 127.0.0.0 0.255.255.255 any
access-list 111 deny   ip 172.16.0.0 0.15.255.255 any
access-list 111 deny   ip 192.168.0.0 0.0.255.255 any
access-list 111 permit ip any any
!-- Default SSM-range. Do not do MSDP in this range
access-list 111 deny   ip any 232.0.0.0 0.255.255.255
access-list 111 permit ip any any
```

http://www.cisco.com/warp/public/105/49.html

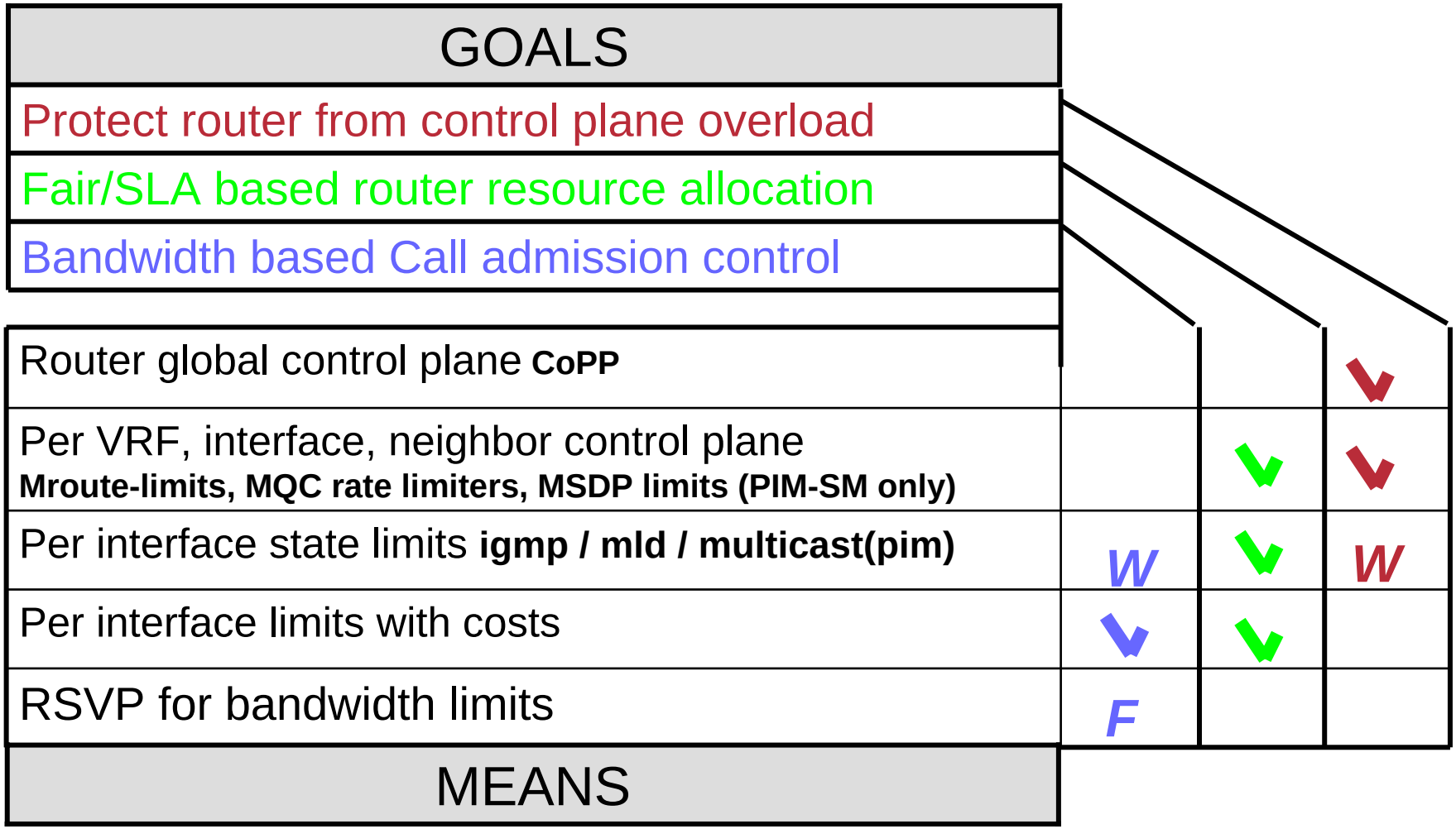
Admission Control



Admission Control
Goals

- **Protect router from control plane overload**
  - **State (HW, memory), CPU**
  - **DoS not to affect non-multicast services**
- **Resource allocation**
  - **Per subscriber (VRF, interface)**
  - **DoS not to affect multicast to other subs.**
  - **Limit subscriber resources to SLA**
- **Call admission control**
  - **Protect bandwidth resources (interfaces, subnets) from congestion**
  - **Content / Subscriber based policies**

Admission Control
Goals and Means (Tools)



↘ - Yes, W – Workaround , F - Future

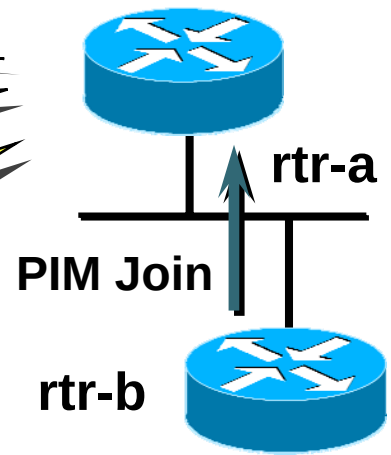
Admission Control
Global / per-VRF Route Limits

ip multicast route-limit <limit> [<threshold>]

```
ip multicast route-limit 1500 1460
```

```
rtr-a> show ip mroute count
IP Multicast Statistics
1460 routes using 471528 bytes of memory
404 groups, 2.61 average sources per group
```

```
%MROUTE-4-ROUTELIMITWARNING :
    multicast route-limit warning 1461 threshold 1460
%MROUTE-4-ROUTELIMIT :
    1501 routes exceeded multicast route-limit of 1500
```



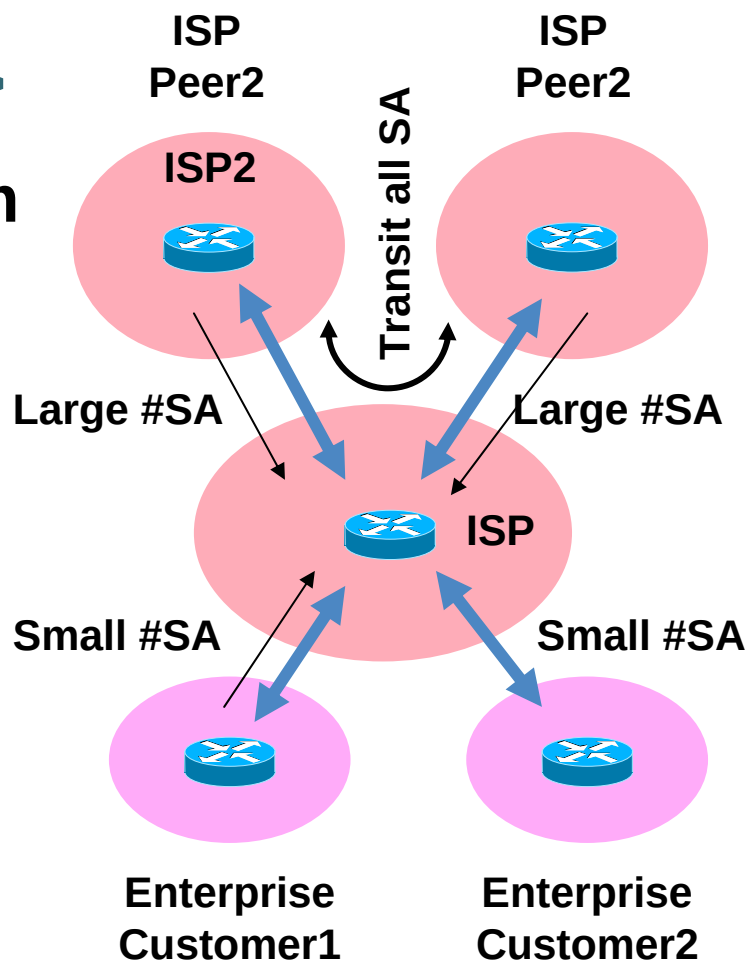
- **No state created beyond <limit>**

  State triggering packets still punted, but discarded

- **Syslog warnings created beyond <threshold>**

Admission Control
MSDP Control Plane

- ip msdp sa-limit <peer> <limit>
- **Limit #SA states accepted from MSDP peer**
- **Simple recommendations:**
  - **Small limit from stub-neighbor (customer)**
  - **Large limit (max #SA in Internet) from transit customer**
    - **If you are transit ISP yourself**
  - **Enterprise MSDP speaker**
    - **Max #SA that won't overload router**



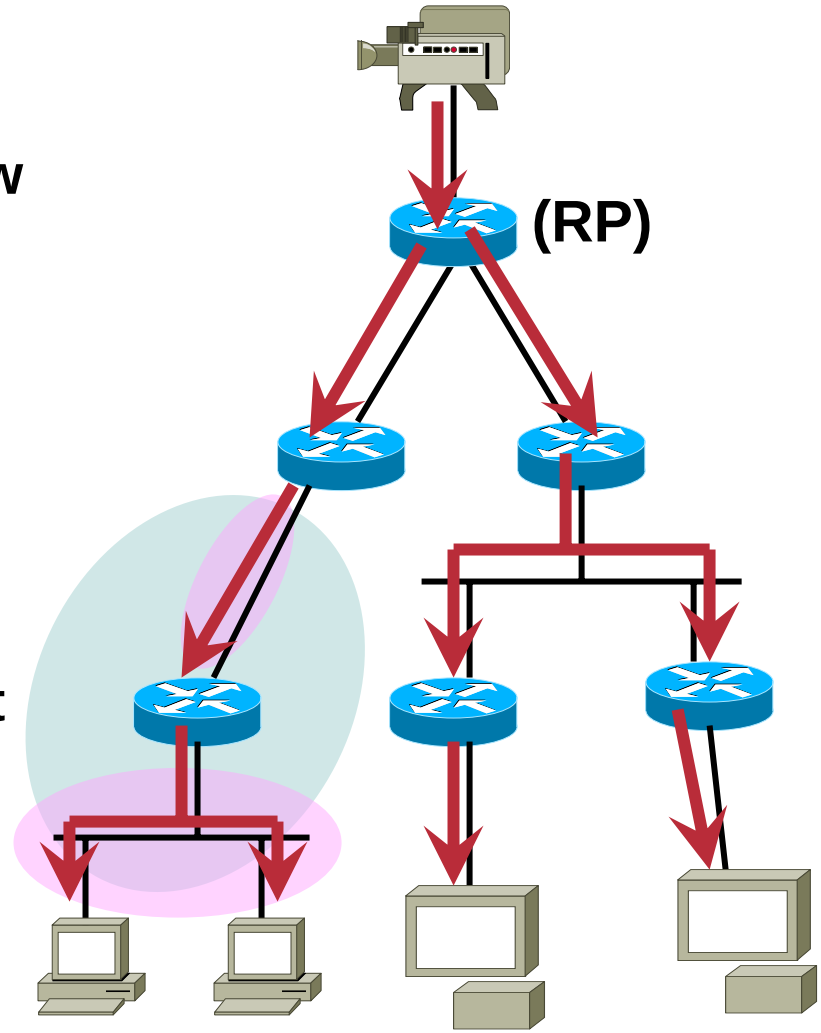
State / Call Admission Control Terminology

- 1 Call =
  1 (TV/radio/market) program / flow
- 1 State ~= 1 Call ?

  SSM: 1 (S,G) state = 1 call

  ASM/PIM-SM: 1 call = (*,G)
  + 1 or more some (S,G)
- Limit state = DoS protection
- Limit calls = service management
- Admission:

  hop by hop –
  permit/deny call for
  whole branch of the tree



Admission Control
Host Receiver Side Admission Control

ip igmp limit <n> [except <ext-acl>]

ipv6 mld limit <n> [except <ext-acl>]

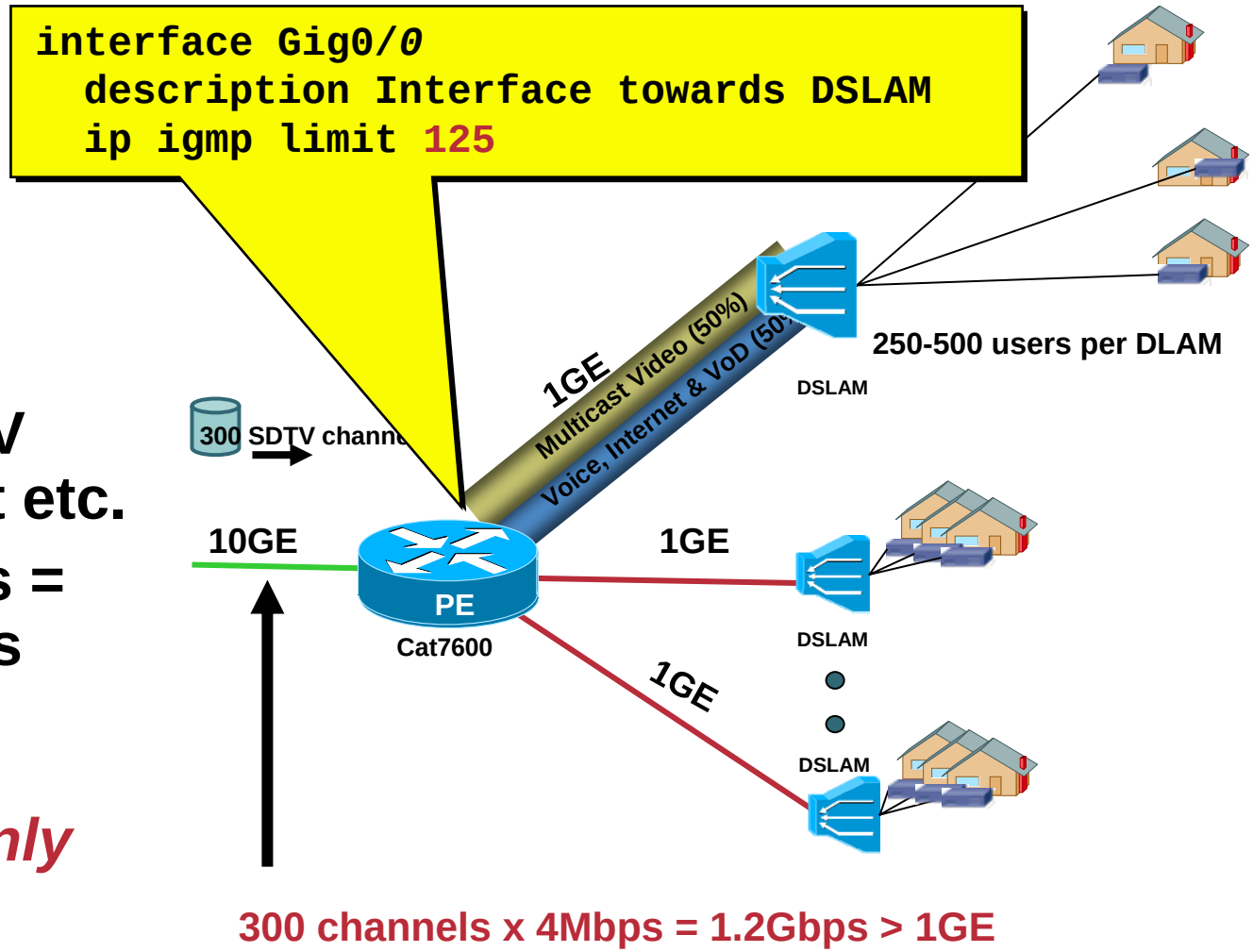
- **Always per interface**
- **Global command sets per-interface default**
- **Counts entries in IGMP cache**

```
ip access-list extended channel-guides
  permit ip any host 239.255.255.254 ! SDR announcements
  deny   ip any any

ip igmp limit 1 except channel-guides

interface ethernet 0
 ip igmp limit 2 except channel-guides
```

Example Usage of igmp Limit Admission Control on Agg-DSLAM Link

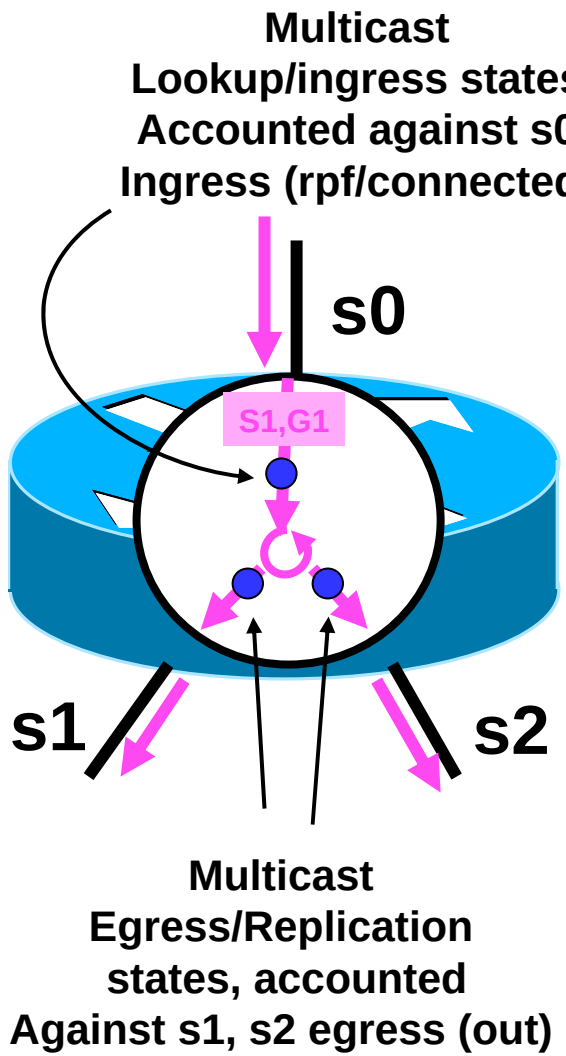
1. 300 SDTV channels
2. 4Mbps each
3. Gbps link to DSLAM
   500 Mbps for TV
   rest for Internet etc.
4. 500Mbps/4Mbps = 125 IGMP states

IGMP/MLD = Receiver side only No PIM



Per Interface mroute Limits

- ip multicast limit
  [rpf | out | connected] <ext-acl> <max>
- Per interface mroute state (PIM/IGMP)
- Input: Rpf,
  connected = (S,G) with S connected
- Output: Out
- Multiple limits allowed per interface
- Each establishes one limiter
- Input / Output state accounted against first limiter permitting state in <ext-acl>



Per Interface mroute Limits

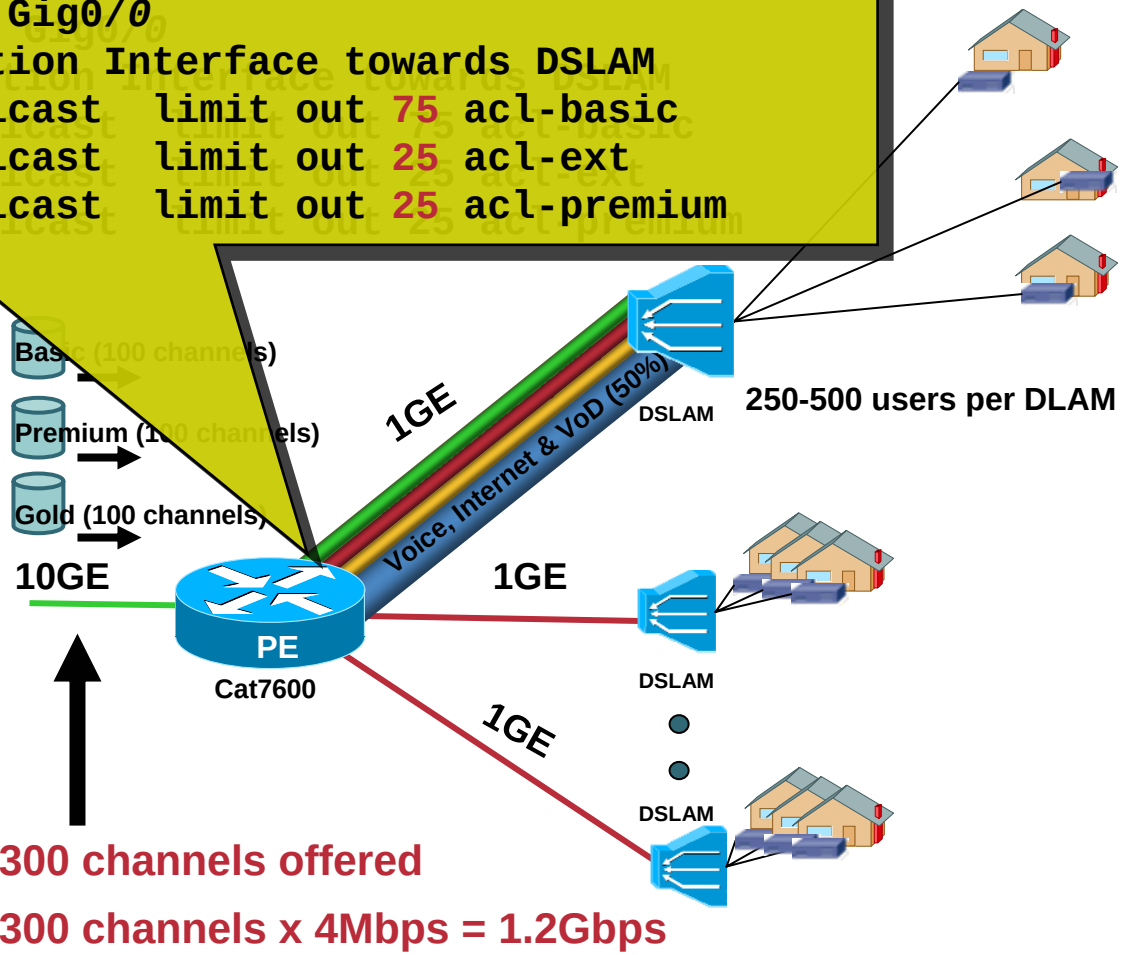
- **Control plane similar to ip multicast boundary**
  - **Same IIF,OIF PIM, IGMP filtering if flow denied**
- **Denied input multicast boundary creates OIF=0 flow state**
  - **Protect best against unwanted packets**
- **Denied input multicast limit does not create state**
  - **Protect against state creation!**
    - **Use CoPP to protect against unwanted packets**
- **Limits state, not calls (*,G) and (S,G) counted !**
- **Emulate boundary with multicast limit!**
  - **ip multicast boundary <ext-acl> out**
  - **ip multicast limit out <not-ext-acl> 0**
    - **Need to invert <ext-acl>**

Example Use of per Interface mroute Limits Admission Control on Agg-DSLAM Link

Generic interface multicast route limit feature with support for Ingress, egress, PIM/IGMP, ASM/SSM

```
interface Gig0/0
 description Interface towards DSLAM
  ip multicast  limit out 75 acl-basic
  ip multicast  limit out 25 acl-ext
  ip multicast  limit out 25 acl-premium
```

1. 300 SD channels
2. 4 Mbps
3. Basic, Extended, Prem. bundles 100 channels ea
4. 500 Mbps for TV
5. 60%/300Mbps Basic
   20%/100Mbps Extended
   20%/100Mbps Premium
6. Basic 75 states
   Premium 25 states
   Gold 25 states



Bandwidth Based CAC

- ip multicast limit cost <ext-acl> <multiplier>
- Cost (<multiplier>) of states in ip multicast limit
- Global configuration, multiple possible
- Use first limit cost with <ext-acl> permitting state
- Usage

  Set multipliers to Mbps/kbps of flows

  Set up address plan to include bandwidth

    Allow to add flows later without changing config

  239.1.X.Y -> X = bandwidth in Mbps (2..20), Y flow

  PIM-SM: count only (*,G): (multiplier=0 for S,G…)

Multicast Call Admission Control :
Cost Factor for per-interface Mroute State Limits

Example CAC use:

1.

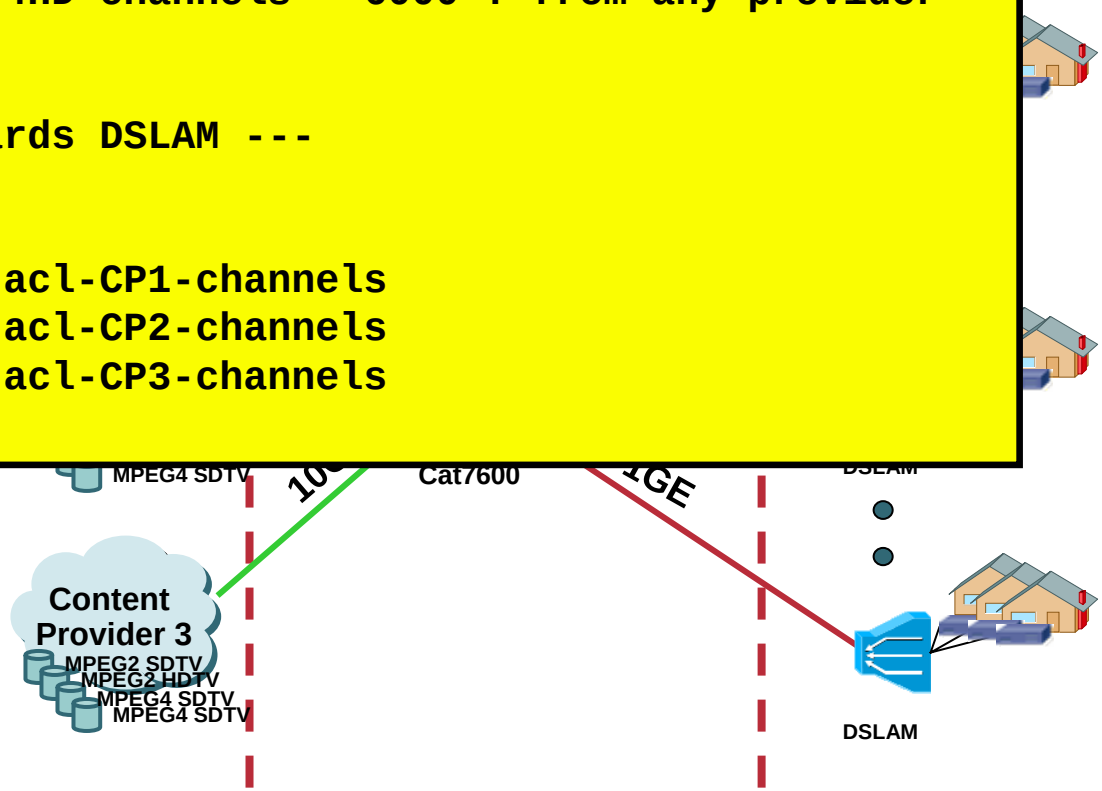
2.

3.

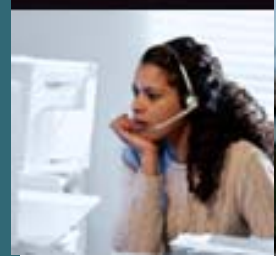
```
! Global
ip multicast limit cost acl-MP2SD-channels   4000 ! from any provider
ip multicast limit cost acl-MP2HD-channels 18000 ! from any provider
ip multicast limit cost acl-MP4SD-channels   1600 ! from any provider
ip multicast limit cost acl-MP4HD-channels   6000 ! from any provider
!
interface Gig0/0
description --- Interface towards DSLAM ---
...
! CAC
ip multicast limit out 250000 acl-CP1-channels
ip multicast limit out 250000 acl-CP2-channels
ip multicast limit out 250000 acl-CP3-channels
…
```

4. 250 Mbps for each CP
   250 Mbps Internet/etc

5. Simply add global costs



Most Simple Recipe for Running IP Multicast ?



Secure Best-Effort SSM Multicast

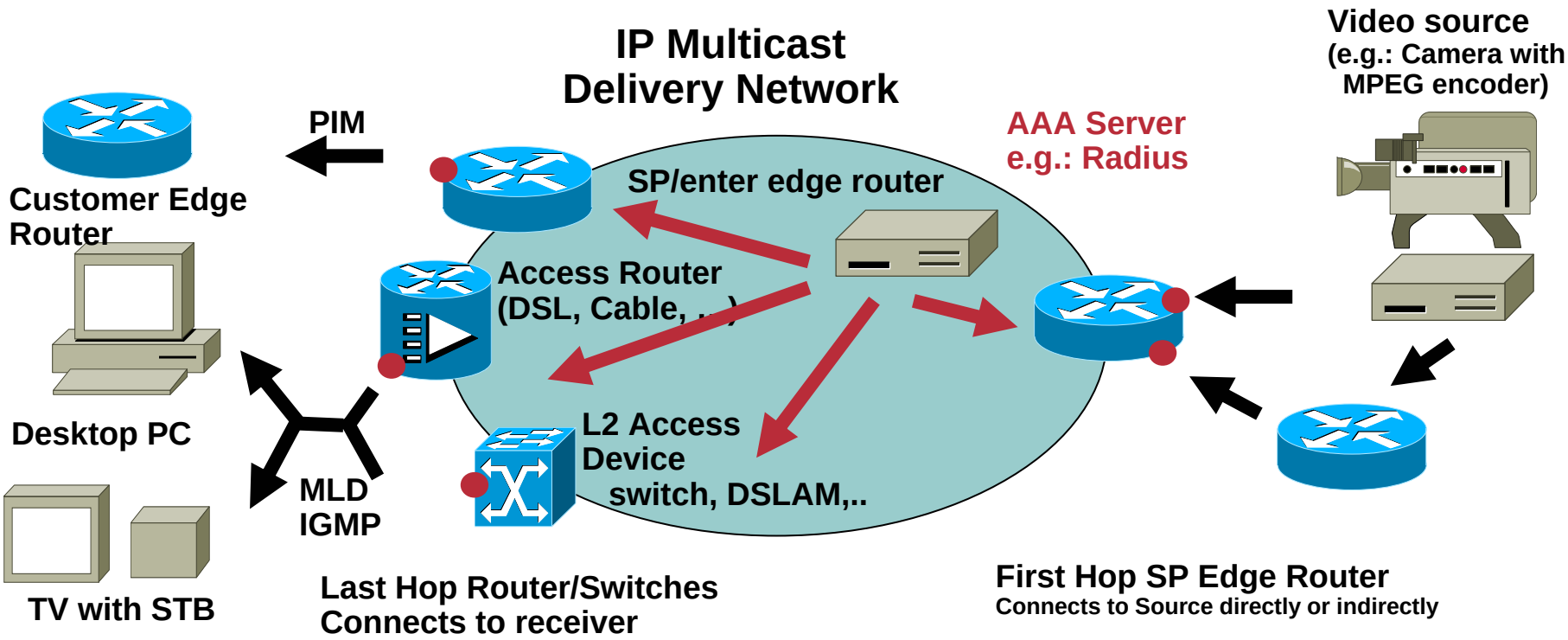
- **Best effort – no business critical application running**
- **Secure – protect to be ~comparable to unicast**
- **Simple config – run without monitoring ?!**

```
ip multicast-routing
ip pim ssm default
no ip dm-fallback
ip multicast route-limit 1000 900
ip igmp limit 100

! All interfaces
interface ethernet 0
 ip pim sparse-mode
```

AAA[1] Integration for IP Multicast

[1] Authentication, Authorization, Accounting

Service Edge and Multicast AAA Integration



- Subscriber and Content Provider Edge interfaces
- Access/admission-control CLI is authorization
- AAA integration:

  Add authentication driven authorization and accounting

Service Edge
Without AAA Support

- ***Assume single subscriber per interface***
- **Configure discussed CLI commands, like:**
  - **ip access-group / ipv6 traffic-filter**
  - **ip igmp access-group / ipv6 mld access-group**
  - **ip multicast boundary [in | out]**
  - **ip pim neighbor filter**
  - **ip igmp limit / ipv6 mld limit**
  - **ip multicast limit [rpf | out]**
- **… to provide subscriber based access and admission control**

AAA models for IP Multicast

- **Authentication**
  - **By interface**
  - **By subscriber (PPP links)**
- **Authorization**
  - ***Manual (previous slide)***
  - **Radius/Tacacs provisioning**
  - **Join/Membership authorization (*FUTURE*)**
- **Accounting**
  - **Dynamic accounting via Radius**
  - ***Netflow support for IP multicast***
  - ***Polling of MIB counter***
  - ***Application level (e.g.: STB)***

Not usually considered to be part of AAA

AAA Models
Radius/Tacacs Provisioning

- **Consider manual global CLI configs**

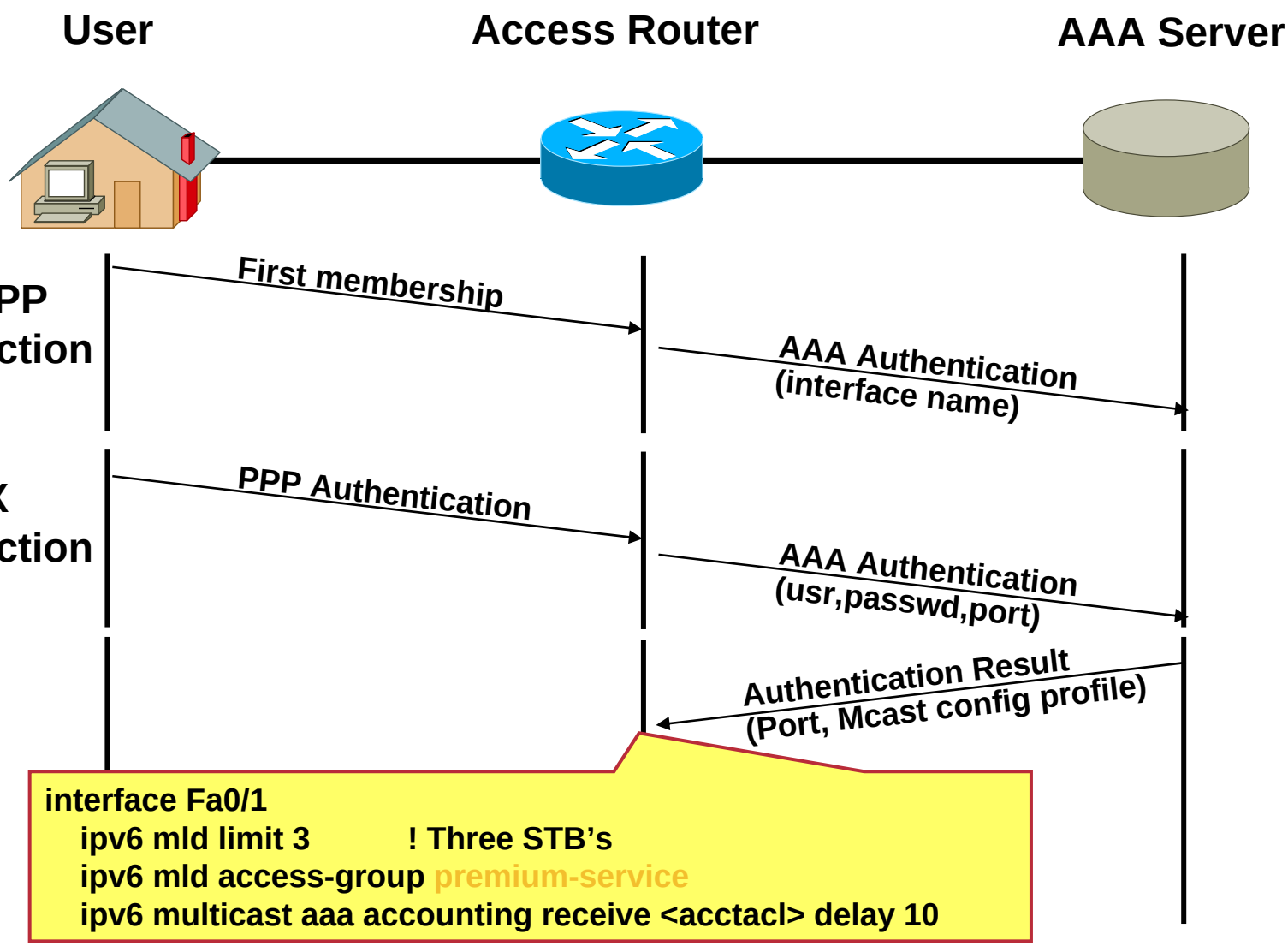
```
! Basic Service
ip access-list standard basic-service
  permit 239.192.1.0 0.0.0.255 ! Basic service channels
!Premium Service
ip access-list standard premium-service
  permit 239.192.1.0 0.0.0.255 ! Basic service channels
  permit 239.192.2.0 0.0.0.255 ! Premium service channels
! Premium Plus Service
ip access-list standard premium-plus-service
  permit 239.192.1.0 0.0.0.255 ! Basic service channels
  permit 239.192.2.0 0.0.0.255 ! Premium service channels
  permit 239.192.3.0 0.0.0.255 ! Premium Plus service channels

! Just all multicast groups
Ip access-list standard all-groups
  permit 224.0.0.0 15.255.255.255
```

AAA Models
Radius/Tacacs Provisioning

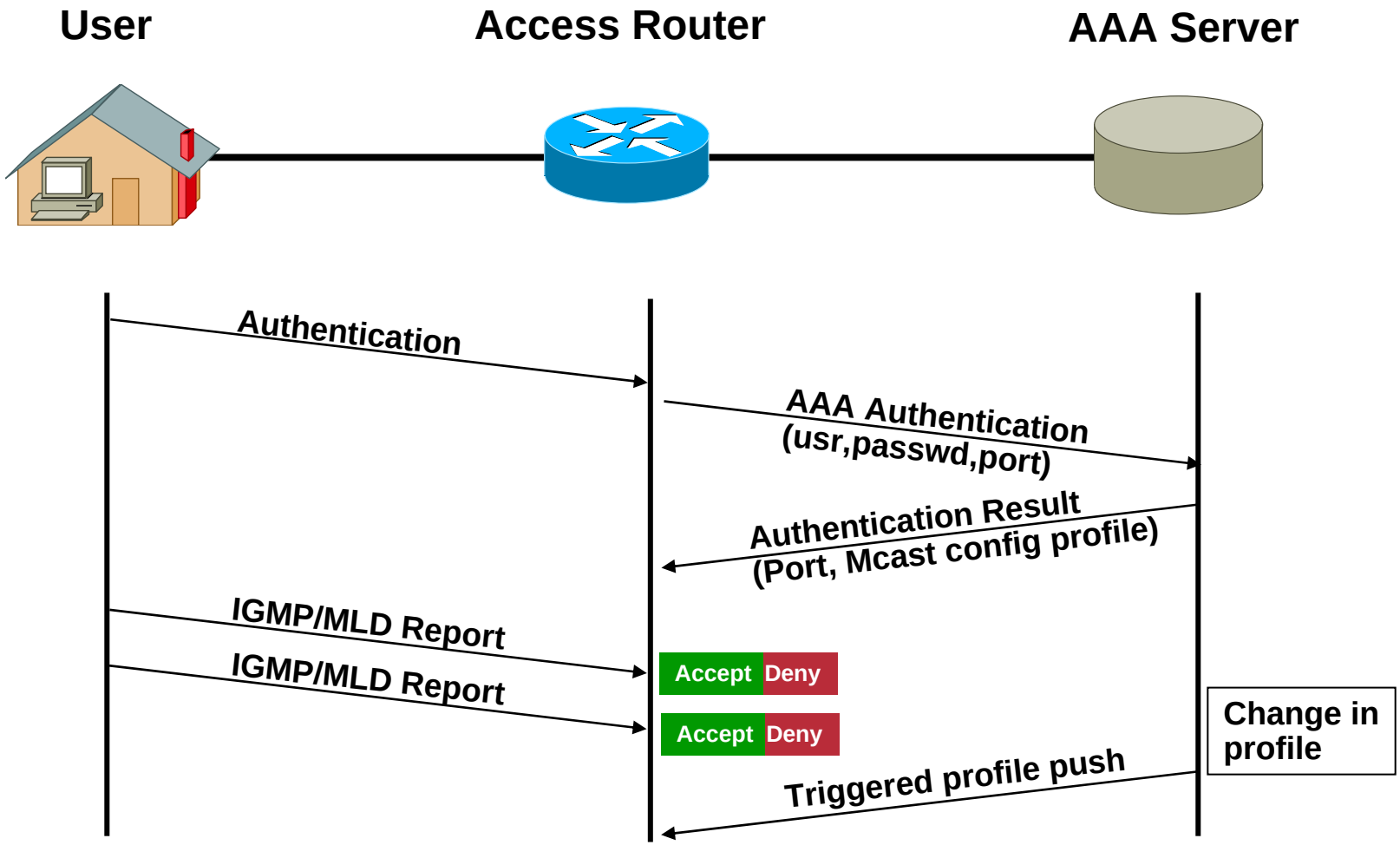
```
! On Radius/Tacacs Server

Username / interface: !(PPP) / non-PPP links
  Cisco:Cisco-avpair = “{lcp}:interface-config =
                       ip igmp limit 3,
                       ip igmp access-group basic-service”
```

- **PPPoX interface support with AAA**
  - **User-ID based authentication. Not specific to multicast.**
  - **Radius server dependent: reuse profiles**
- **Multicast AAA**
  - **aaa authorization multicast default [method3 | method4]**
  - **Trigger Radius authentication after first IGMP/MLD join.**
  - **Authenticates user via interface name**

AAA Models
Authentication with Radius Provisioning



AAA Models
Changing User Profiles from Radius/Tacacs Provisioning



AAA Receiver Accounting

- **aaa accounting multicast default [start-stop | stop-only] [broadcast] [method1] [method2] [method3] [method4]**

  Set global parameters for accounting

- **ipv6 multicast aaa account receive access-list [throttle seconds]**

  Enable accounting on interface

- **Generate Radius acct. records for multicast flows**

  When first joined on an interface (START)

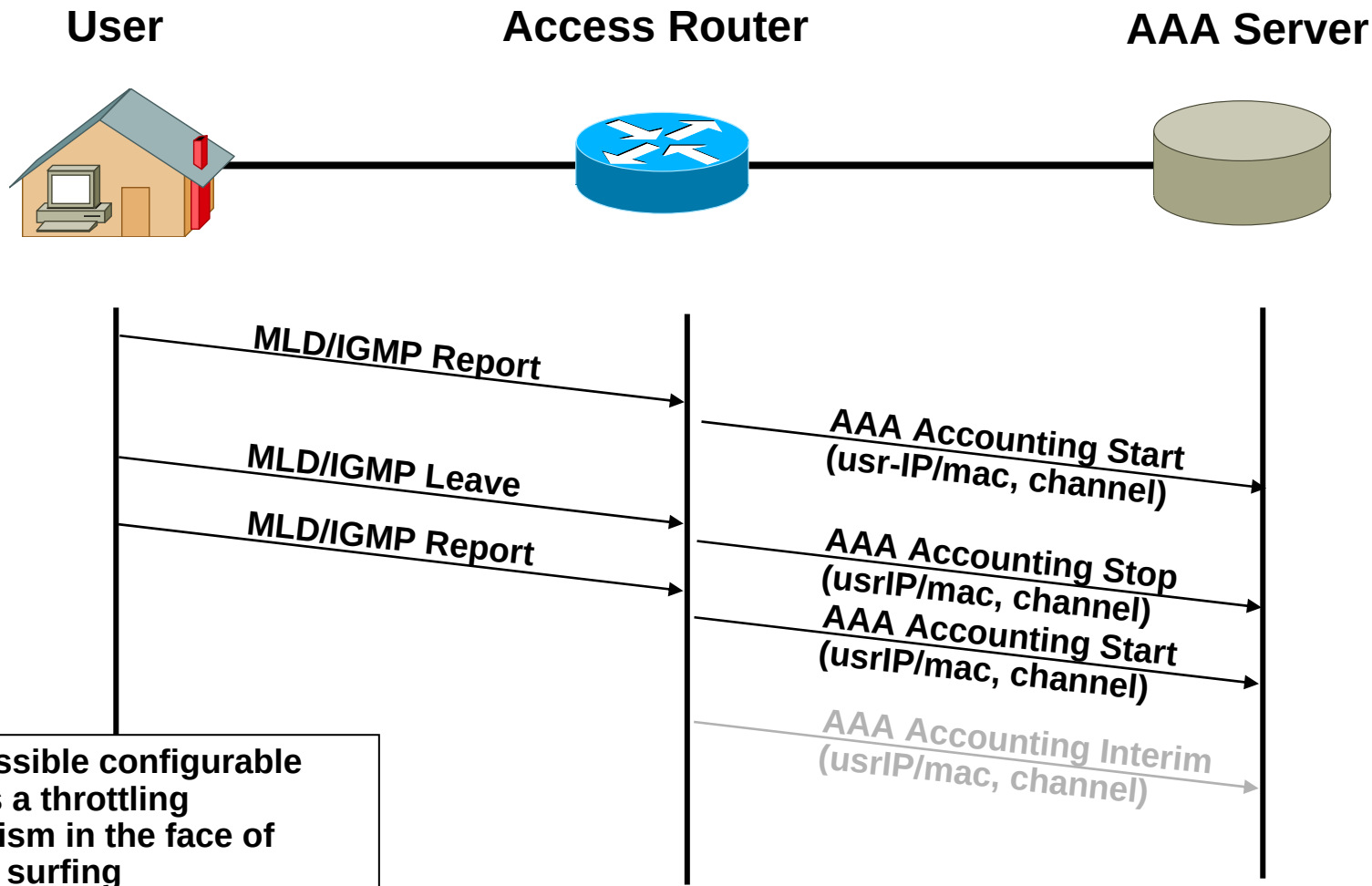
  When stopped being forwarded on interface (STOP)

  FUTURE: Periodically, with counters

- **Avoid many accounting record during zapping**

  Send START record only throttle-seconds after join

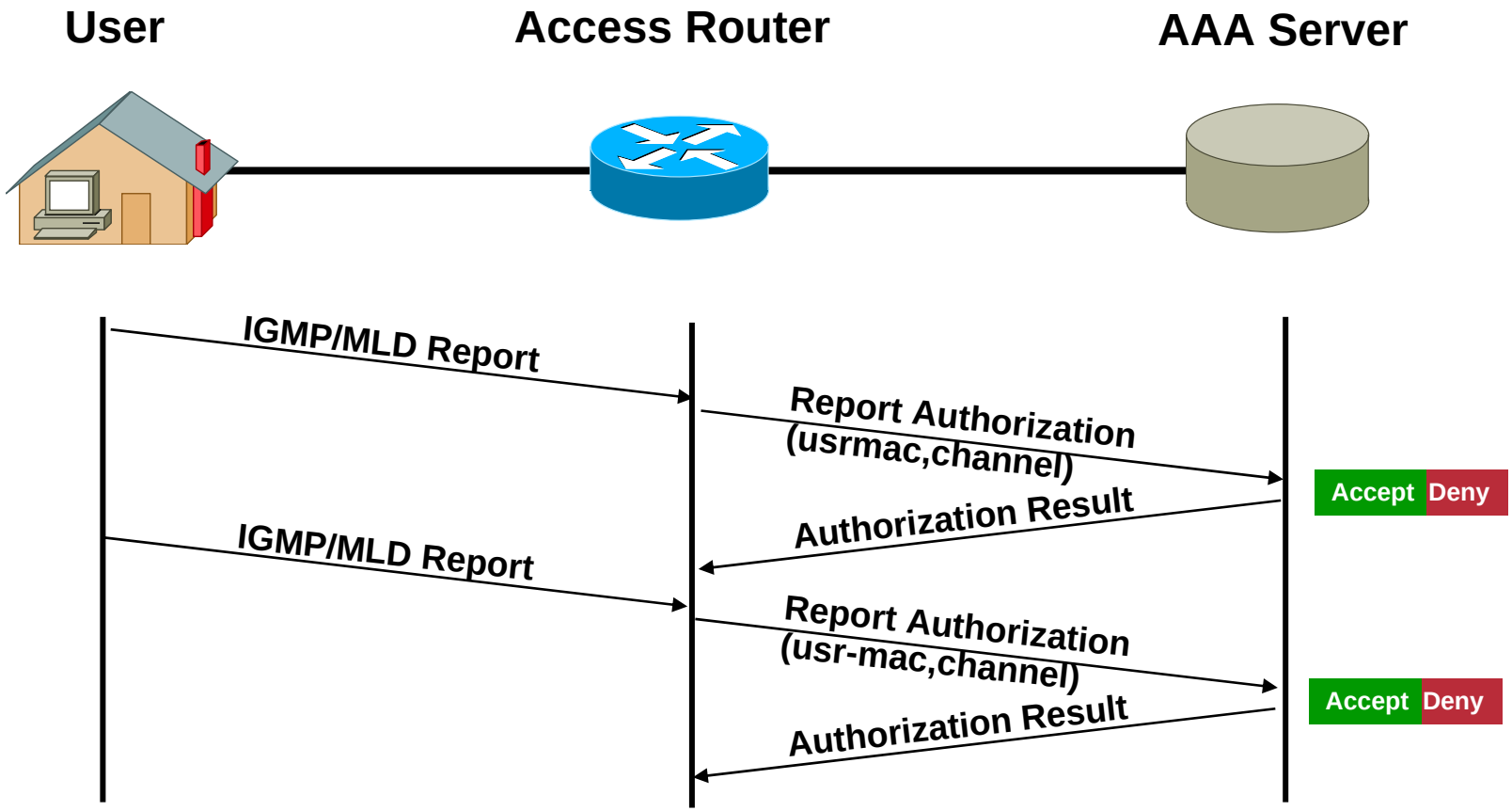
AAA Receiver Accounting



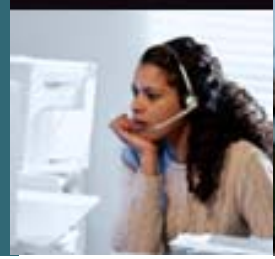
AAA Models *FUTURE*
Per Join/Membership Authorization

- **Allows dynamic AAA server based policies**
  - **More flexible and expensive than provisioning based**
  - **Frequently changing policies (PPV,…)**
  - **Admission control: With tracking of existing memberships**
  - **Scalability via cache-timeout in AAA server replies**
- **Potential to combine models:**
  - **Whitelist: AAA provisioning permits always allowed services**
  - **Greylist: per join/membership authorization only invoked for requests not permitted by whitelist**
    - **E.g.: PPV, ..**
  - **Maximizes scalability**

AAA Models *FUTURE*
Per Join/Membership Authorization



Firewalls



Firewalls and Multicast
Overview

- **Firewall function is not NAT**
  - **Firewall and NAT often collocated**
  - **IOS: Source NAT and limited src/grp NAT**
- **IOS Firewall: specific firewall feature set**
  - **No multicast support requirements identified**
  - **Coexistence good enough!**
- **Firewall devices/appliances: (PIX, ..)**
  - **Add IP multicast routing to device**
  - **And similar access control as IOS**

Cisco IOS Firewall

- **Unicast**
  - **Mostly to avoid unwanted “receiving”/”external connections”**
  - **Identify TCP/UDP/RTP/.. Flows by examining control plane flows (TCP, RTSP, FTP, HTML, ..)**
  - **Dynamic permitting flows based on policies**
  - **Ignores multicast traffic (passes through)**
- **Multicast**
  - **No important control plane driven multicast flow setup identified / required by customers (so far)**
  - **Use multicast access-control to permit multicast applications:**
    - **ip multicast boundary, …**
  - **Safe because (you should know this now):**
    - **Multicast flows stateful, explicit join**
    - **Full control of flows with existing control mechanisms**

PIX Firewalls
Overview

- **PIX Firewall pre v7.0**
  - **IGMPv2 proxy routing only (for edge PIX)**
- **PIX Firewall v7.0**
  - **PIX is full IGMP/PIM router**
    - **Full features (ported from IOS)**
    - ***Not all functions tested yet / supported***
    - **Obsoletes solutions like**
      - **GRE tunnels through PIX**
      - **Third-party DVMRP-only firewall**

PIX Multicast
Feature SUMMARY (Incomplete)

- **Multicast Support in PIXOS 7.0**
- **IGMP Support**
  - **Stub multicast routing – IGMP Proxy Agent**
  - **IGMPv2, access group, limits**
- **PIM Support**
  - **PIM Sparse Mode, Bidirectional, SSM**
  - **DR Priority**
  - **Accept Register Filter**
  - **Multicast Source NAT**
- **515, 525, 535 and ASA platforms**

PIX Multicast
Candidate Future Features

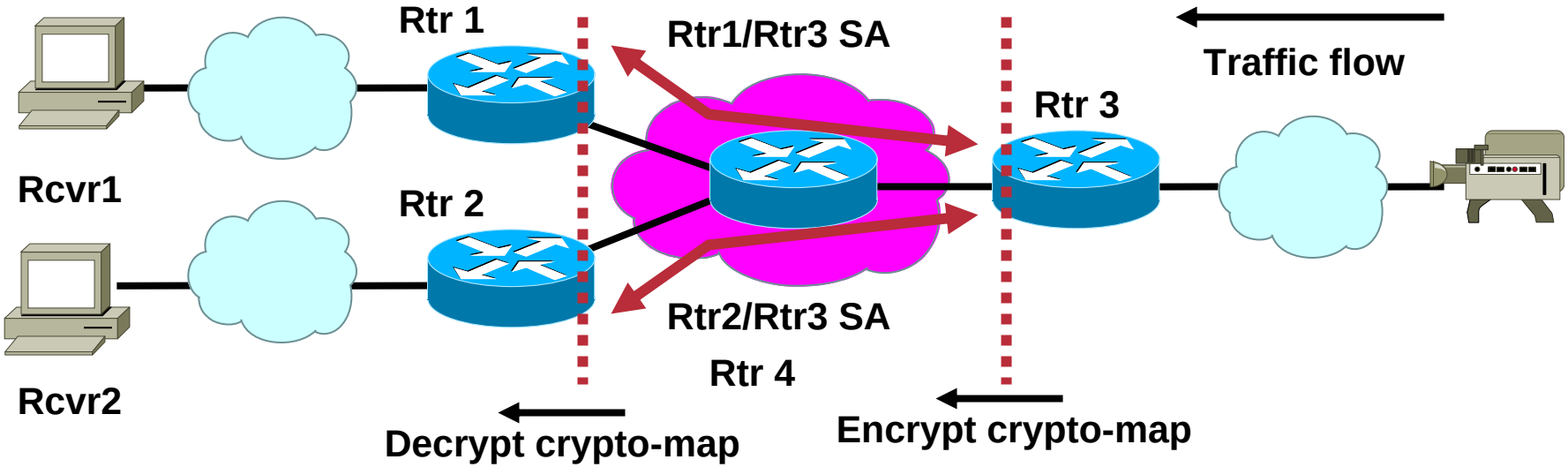
- **Planned Multicast Support in PIXOS 7.2**
  - **Multicast Boundary with autorp filter**
  - **PIM Neighbor filters**
  - **PIM Bidir Neighbor filters**
- **Future Support ?**
  - **IGMPv3/SSM**
  - **IPv6 Multicast**
  - **Stateful MSDP inspection**
  - **MSDP**
- **FSWM 3.1 will support multicast similar to PIXOS 7.0**

IPsec and Multicast

Multicast and IPsec concepts

- **IPsec p2p tunnel interfaces (12.3(14)T/12.3(2)T)**
  - **Permits to encrypt IP multicast traffic**
    - **avoids RPF issues with crypto-map based IPsec**
- **“Secure Multicast” (12.4(6)T)**
  - **Transparent “tunnel mode” en/decryption**
    - **Inhibits need for overlay network**
    - **Allows to apply IPsec to both multicast data traffic and control plane traffic (e.g.: PIM)**
  - **GDOI--Key distribution mechanism (RFC3547)**
    - **Manual keying still available**

Legacy behavior for IPsec multicast



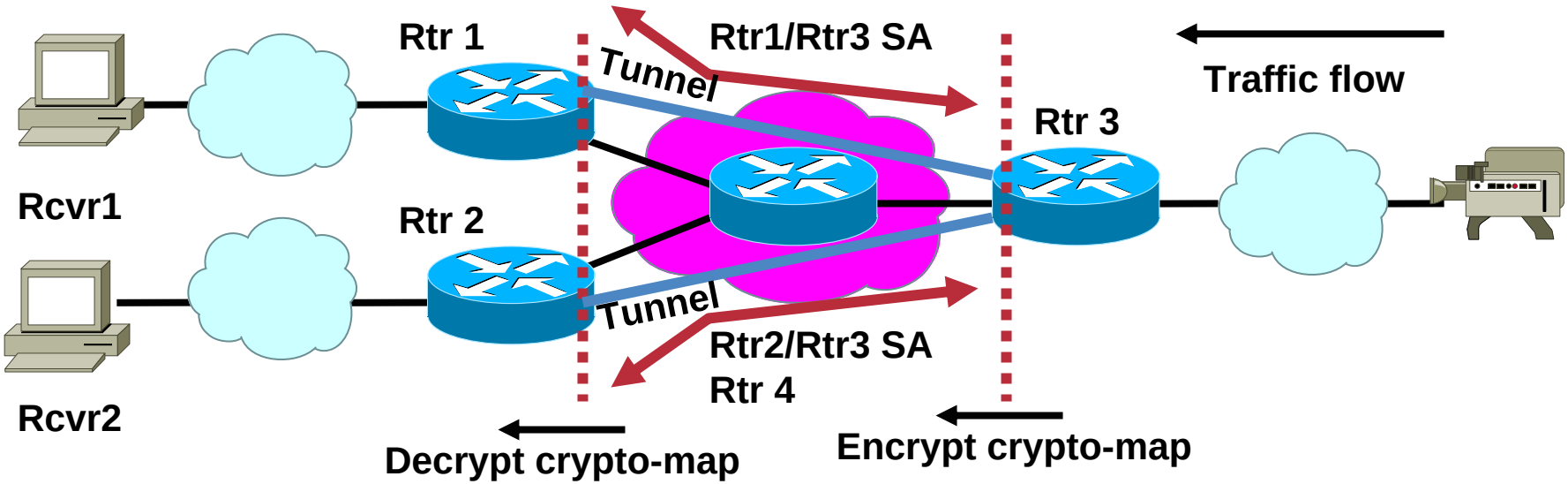
- Not supported for multicast, broken!
- Security associations (SA) - Rtr1/Rtr3 and Rtr2/Rtr3.
- “tunnel mode” (unicast) encapsulated Rtr3->Rtr1, Rtr3->Rtr1
- Broken because (lots of reasons):

  Rtr1/Rtr2 do not see Rtr3 as PIM neighbors, but Rtr4

  Sending PIM joins to Rtr4 ineffective, Replication on Rtr3 ineffective, …

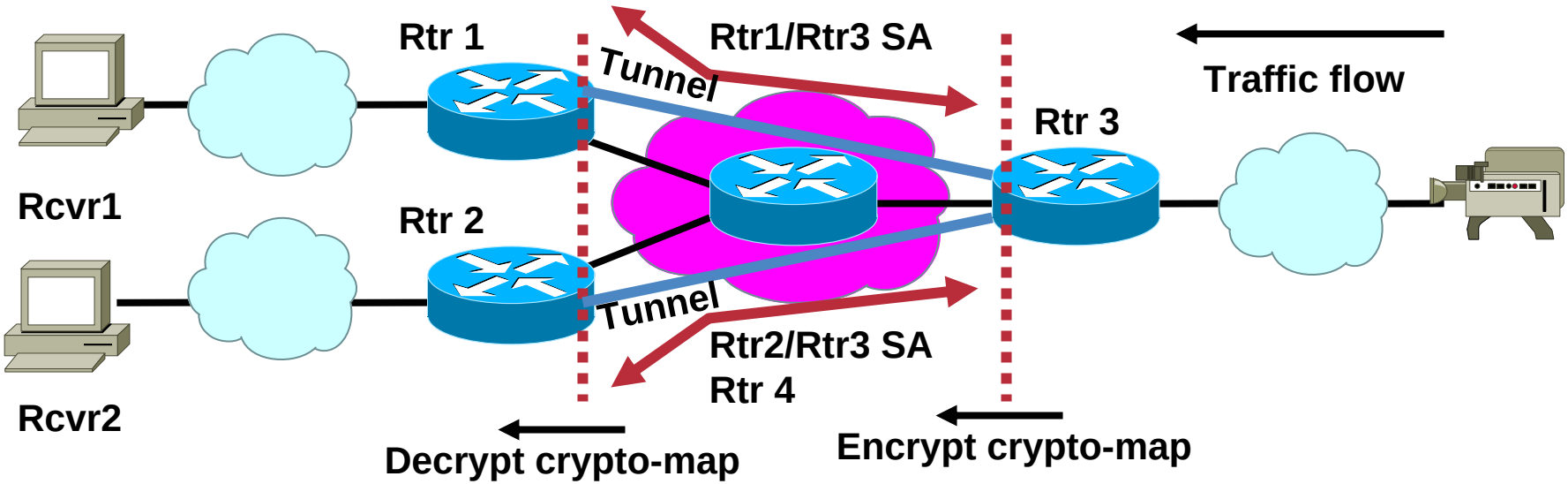
  Would require hacks/NBMA mode interaction on Rtr3, etc…

Working P2P tunneling multicast with IPsec Tunnels with explicit tunnel interfaces



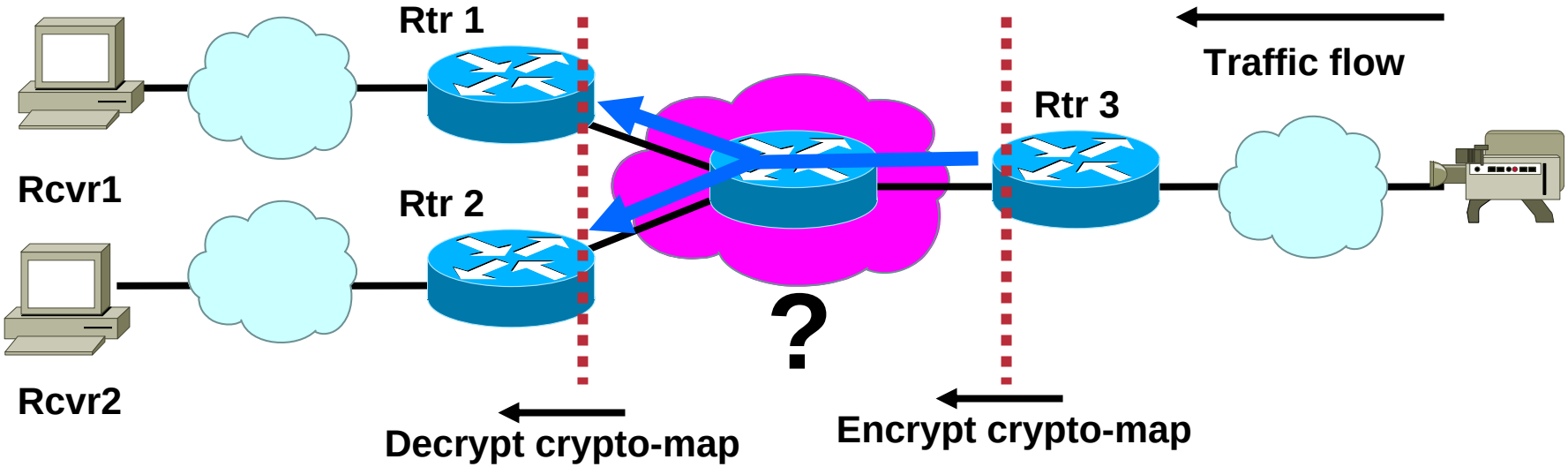
- **GRE tunnel Rtr1/Rtr3 and Rtr2/Rtr4**
  - **Apply IPsec encryption only to GRE (unicast) traffic**
  - **Used by (for example) DMVPN solution**
  - **Also scales: Hub (Rtr3) can use single multipoint tunnel interface to support hundreds/thousands of spokes**

Working P2P tunneling multicast with IPsec Use tunnels with explicit tunnel interfaces



- **New: IPsec tunnel interface**
  - **Looks like GRE tunnel to IP multicast**
  - **Does not replace GRE for DMVPN**
    - **GRE beneficial for NHRP operations**
    - **Scalability of multipoint GRE**
  - **Beneficial for few IPsec tunnels, forwarding performance and interoperability with 3rd party IPsec equipment.**

Secure multicast
How to use multicast in core ?



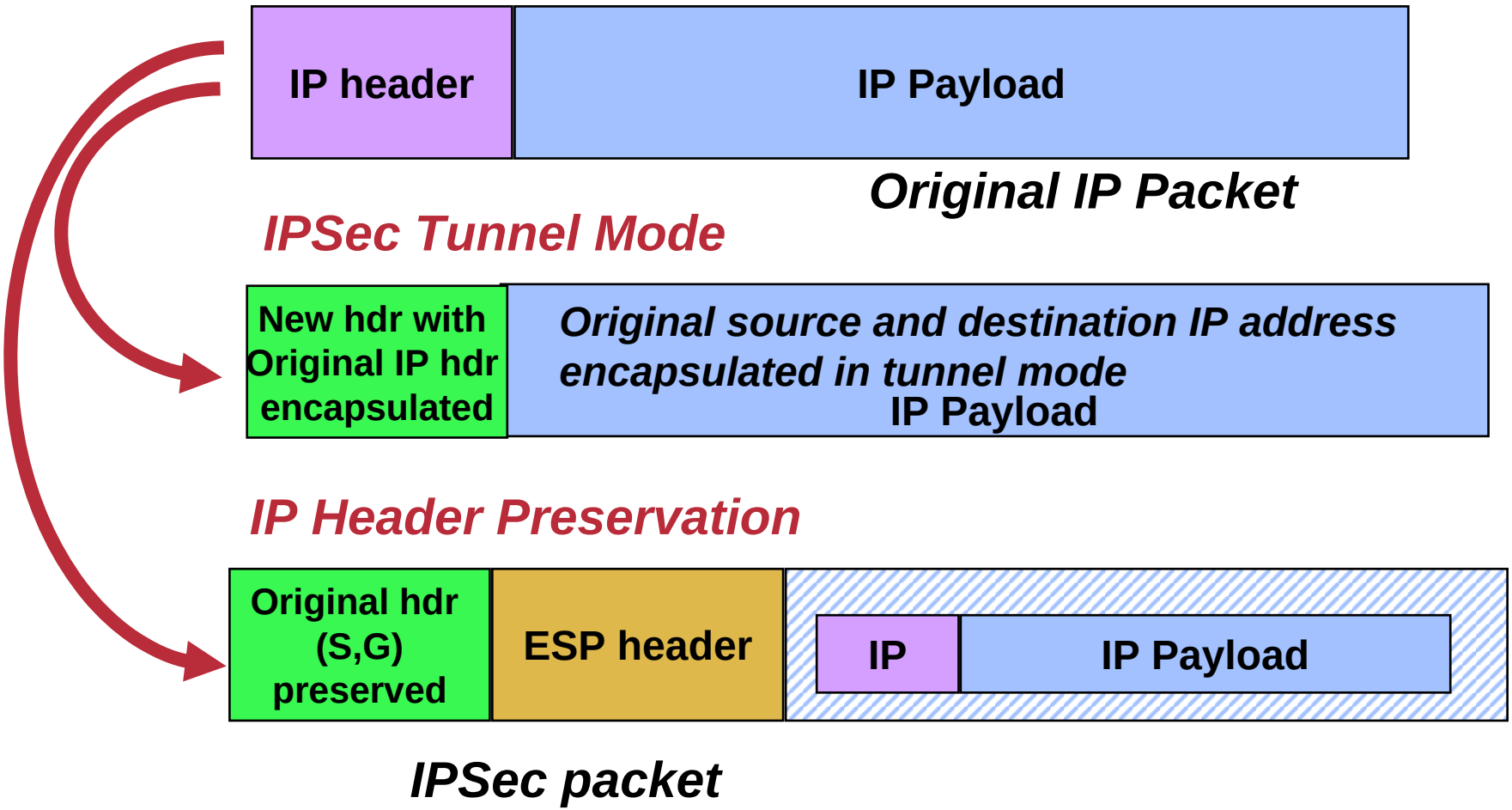
- **IPsec “Tunnel mode”: changes (S,G) – creates multicast overlay problem -> requires MVPN signaling or similar**

  Header preservation can avoid this

- **Need SA for multicast packets between source/receiver routers “group of nodes”)**

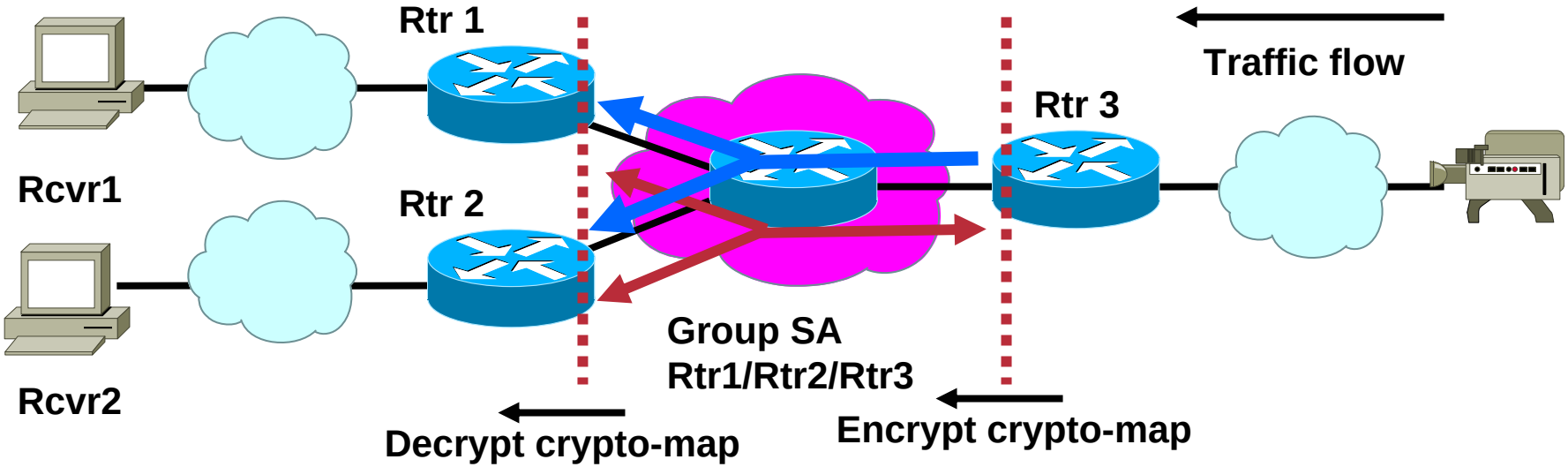
  Manual keying group membership / scaling issues -> GDOI !

IPsec Tunnel Mode: IP Header Preservation



Preservation copies/maintains Source, Destination, options, DSCP, …
Not maintained: IP header protocol type (obviously!)

Secure multicast
Encrypt with multicast across core



- **In IOS images with "Secure Multicast" feature support, multicast packets receive "Header Preservation" in tunnel mode**
- **Can now successfully use crypto-maps (no tunnel interfaces required) to pass multicast encrypted across core**
- **Use with:**

  MVPN: en/decrypt on PE or CE routers !

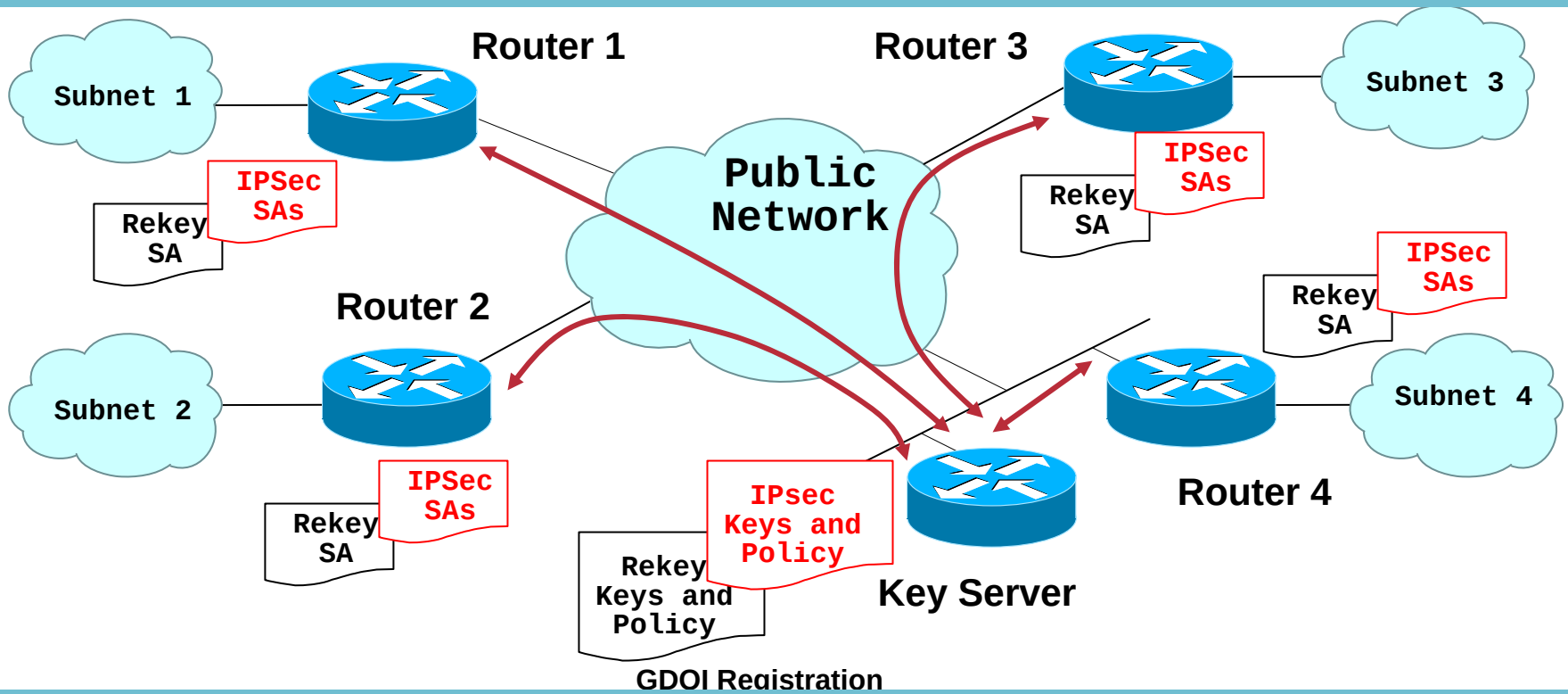
  Non-MVPN: E.g.: Enterprise, unsecure backbone links, …

Dynamic group SA keying - GDOI

- **Single/manual configured key between all encrypting/decrypting routers**
  - **How to manage keys, membership? Re-keying ?**
- **GDOI: Dynamic Group-SA protocol (RFC3547)**
  - **Group-key equivalent to PKI (p2p SA)**
  - **Client-Server protocol:**
    - **Encrypting/decrypting nodes (routers/host) = clients**
    - **Server: Key server, managing members**
    - **IOS implements both client and server side**
  - **Scalable/dynamic re-keying: Can use multicast to distribute updated keys !**

GDOI example

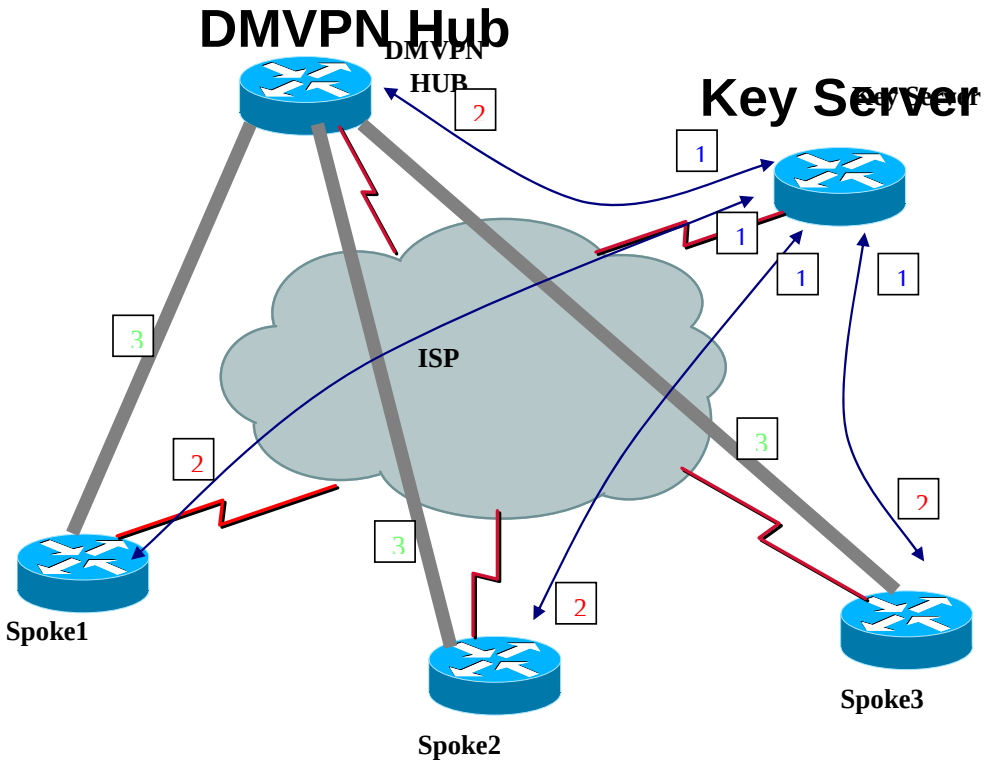
- **Each router Registers with the Key Server. The Key Server authenticates the router, performs an authorization check, and downloads the policy and keys to the router.**



Each VPN CPE

- Registers to the GDOI key server to “receive” IPsec SAs
- Encrypts/Decrypts packets matching the SA
- Receives re-key messages, either to refresh the IPsec SAs or eject a member

Application Scenario:
Integration of GDOI with DMVPN



Benefit: **Using Secure Multicast / GDOI functionality in DMVPN network, the delay from IPSec negotiation is eliminated**

Note : Multicast traffic will be still forwarded to Hub for any spoke to spoke even with this deployment.

1. DMVPN Hub and spokes are configured as Group Member (GM)
2. All Group members register with the Key Server (KS)
3. A spoke to hub tunnel is established using NHRP
4. Spoke sends a NHRP resolution request to the Hub for any spoke-spoke Communication
5. Upon receiving NHRP resolution reply from the hub, the spoke sends traffic directly to other spokes with group key encryption

Secure PIM Control Traffic with IPSec

- **Encrypt/Authenticate PIM Packets**
  - **Crypto map for 224.0.0.13 (PIM Control Messages)**
  - **Use either IPSec options**
    - **Hash Functions: MD5, SHA1**
    - **Security Protocols: Authentication Header(AH), Encapsulating Security Payload (ESP)**
    - **Encryption Algorithms: DES, 3DES, AES**
    - **Recommended IPSec Mode: Transport**
    - **Recommended Key method: Manual ?**
  - **IPSec AH recommended in PIM IETF drafts**

Secure PIM Control Traffic Example

```
access-list 106 permit ip 0.0.0.0 255.255.255.255 host 224.0.0.13

crypto ipsec transform-set pimts ah-sha-hmac
   mode transport

crypto map pim-crypto 10 ipsec-manual
 set peer 224.0.0.13
 set session-key inbound ah 404 bcbcbcbcbcbcaaaa
 set session-key outbound ah 404 bcbcbcbcbcbcaaaa
 set transform-set pimts
 match address 106

interface Ethernet0/0
   crypto map pim-crypto
```

Secure multicast summary
Key Application Scenarios

Key Use Case	Customer	Features
Encryption of IP packets sent over Satellite Links	Organizations who wish to secure video communications through use of BB satellite	Hardware Acceleration support Native Multicast Encryption
Secured Multicast VPN	MPLS VPN Service Provider customer who wish to have multicast services between multiple sites of a customer VPN	Security for mVPN packets DoS protection for mVPN PE CE-CE protection for Multicast
Reduce delays in Spoke-Spoke DMVPN network	DMVPN Enterprise customers who are deploying voice and wish to reduce the delays in setting up voice calls between spokes	GDOI with DMVPN Instant spoke-spoke connectivity
Secure PIM Control Traffic	Enterprise financial customers who wish to secure PIM control traffic in their network	PIM control packets encryption

Conclusion

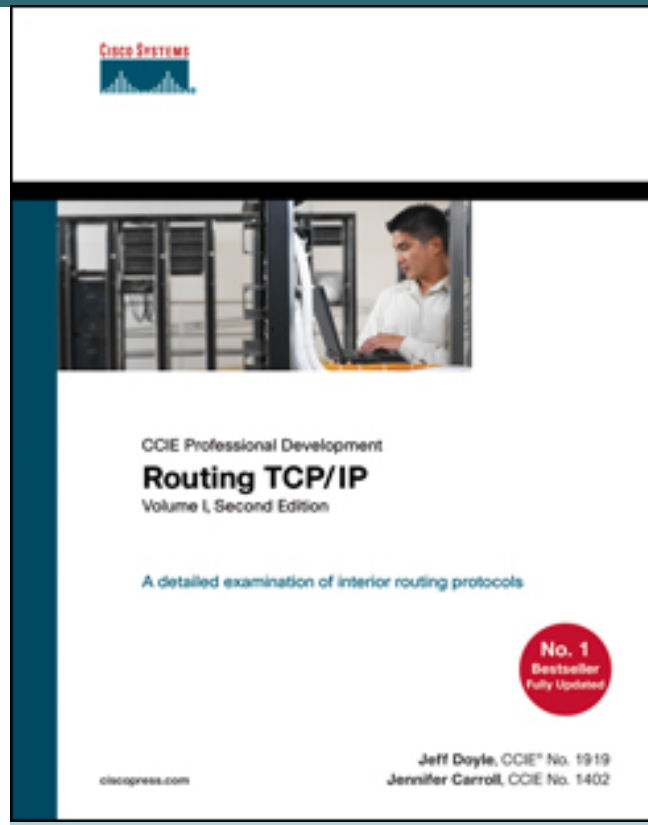
Conclusion
Multicast and Security

- **Multicast is different from unicast !**
  - **Why ? Remember ? … states, replication, joins, unidirectional..**
- **Rich framework of IOS CLI commands for control**
  - **Centered around controlling protocol operations, states (multicast) or policing packets (MQC, CoPP, - same as unicast)**
  - **Can well provide “protected” service/sla**
  - **No “simple” protocol security against DoS (-> IPsec)**
- **PIX with “full”(PIM) IP multicast routing**
- **Emerging solutions with multicast**
  - **AAA, IPsec (protect PIM or multicast data)**

Questions ?

Recommended Reading

- **Continue your Cisco Networkers learning experience with further reading from Cisco Press**
- **Check the Recommended Reading flyer for suggested books**



Available Onsite at the Cisco Company Store

Complete Your Online Session Evaluation

- **Win fabulous prizes; Give us your feedback**
- **Receive ten Passport Points for each session evaluation you complete**
- **Go to the Internet stations located throughout the Convention Center to complete your session evaluation**
- **Drawings will be held in the World of Solutions**

  Tuesday, June 20 at 12:15 p.m.

  Wednesday, June 21 at 12:15 p.m.

  Thursday, June 22 at 12:15 p.m. and 2:00 p.m.