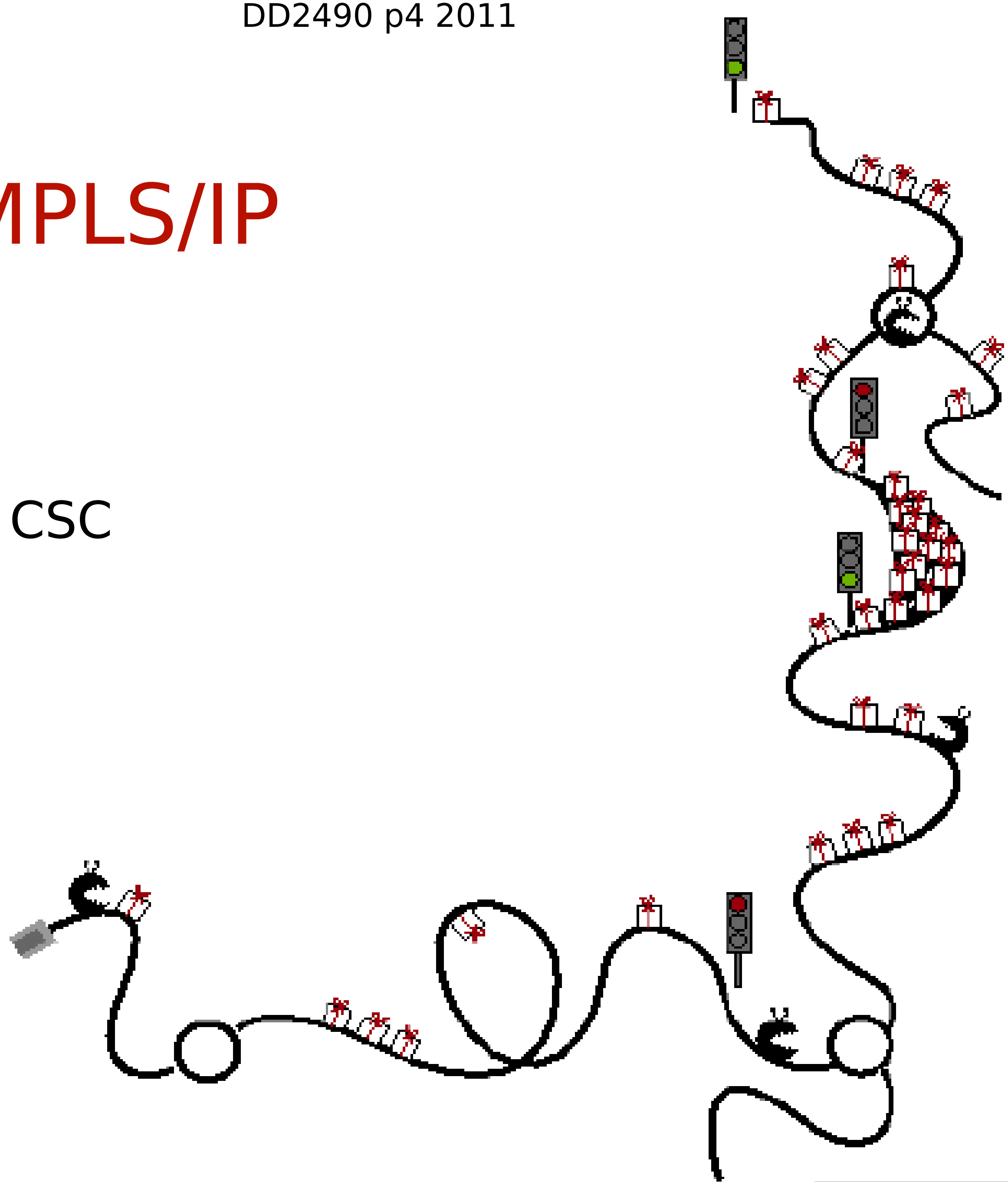
Routing and MPLS/IP



Olof Hagsand KTH CSC

Literature

- Lecture slides and lecture notes (on web)
- Reference
  - JunOS Cookbook: Chapter 14

Background

- MPLS - Multiprotocol Label Switching
- Originally thought to simplify IP forwarding
  - Small label lookup instead of longest prefix match
- Roots in ATM (Asynchronous Transfer Mode)
  - Early 90s, most telecoms thought ATM would take over all data- and tele-communication
- MPLS/IP was standardized in IETF in mid 90s.
- *GMPLS* can be used for optical networking such as management of wavelengths: ”lambdas”
- *MPLS/TP* is currently being standardized
  - Transport Profile
  - MPLS/TP is independent of IP and for non-signalled low-level optical networks

MPLS Advantages

Originally, the motivation was speed and cost.

But routers does IP lookup in hardware at very high speeds.

Current advantages:

- Label switching can be used for *traffic engineering*
  - –Aggregating a class of traffic
  - –Guarantees: Allocation of resources
  - –Constrained routing: load, bw, etc.
- Labels can be used to forward using other fields than destination address
- Label switching can be used to support VPNs – virtual private networks

Where is MPLS used?

- MPLS is used as a tunneling technique within an operator's internal IP network
  - Tunneling characteristics - traffic is isolated
  - VPNs
  - Traffic engineering - control bandwidths and links
- MPLS is *not* used
  - In traditional *enterprise* networks
  - Between operators (inter-domain)

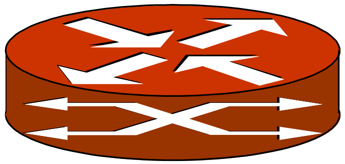
Why MPLS?

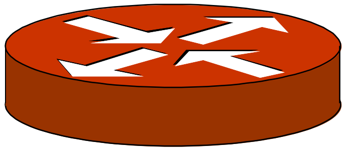
- MPLS gives a simple tunneling mechanism integrated with IP
- Another IP-based tunneling protocol could give the same service
  - IP in IP
- But MPLS has a nice toolbox and is ”easy” to configure
- Alternatives
  - Pure IP networking: ”manage tunnels yourself”
  - Provider backbone bridging (IEEE 802.1ah)

MPLS Terminology

MPLS uses some new terminology (MPLS ~ IP)

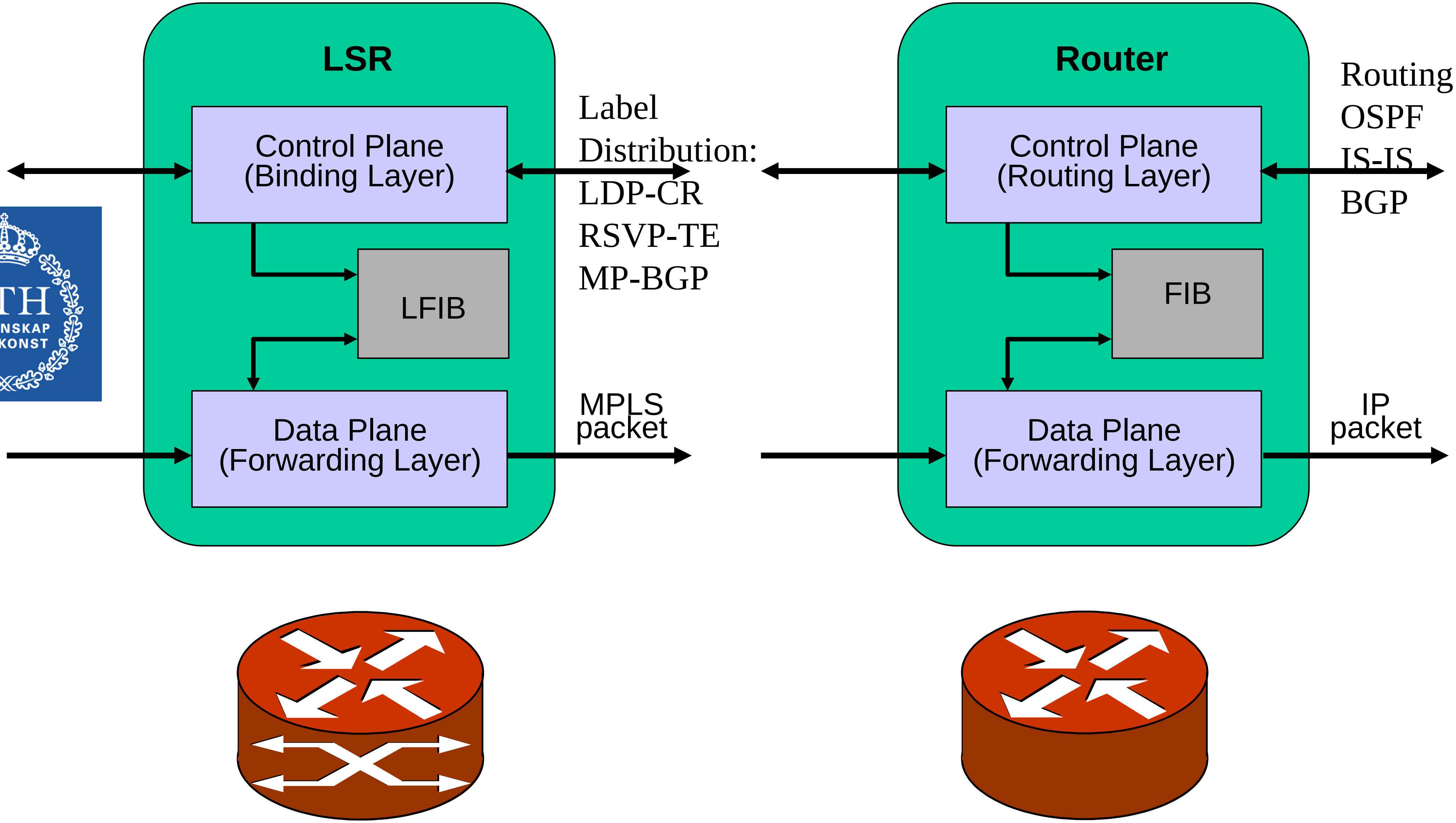
- LSR: Label Switching Router ~ Router





- LER: Label Edge Router ~ Border router
  - Alternative: PE - Provider Edge / CE – Customer Edge
  - Also: Egress/Ingress/Transit LSR
- LSP: Label Switched Path ~ Tunnel
- FEC: Forwarding Equivalence Class ~ Flow
- Label distribution ~ Routing
- LFIB: Label Forwarding Information Base ~ FIB

Control and Data Planes



Labels

- A label is an integer identifying a FEC (a flow).
- You cannot have globally or network- unique labels
  - Too complex to negotiate
  - Too large labels
- Labels are unique only between two nodes
- Labels 0-1048575.
  - 0-15 reserved by the IETF.
- Labels change at each node as a packet traverses a path
- You can set labels manually(worse than static routing), or use label distribution
- Example of Label Forwarding Information Base LFIB (MPLS forwarding table):

20-bit label

In-Interface	in-label	op	out-label	out-interface
if 3	99	swap	203	if1
if 3	333	swap	978	if2

Encapsulation

20-bit label	3-bit TC	1-bit stack	8-bit TTL

Eth hdr	Label hdr	IP hdr	IP payload

- MPLS uses a 32-bit *shim* header:
  - –Label: Value for table lookup in router
  - –TC: Traffic class field, can be used as class-of-service for QoS
  - –Stack: Indicates that the bottom of a stack of labels has been reached
  - –TTL: Time To Live (resembles IP TTL)
- Shim headers may be concatenated into *stacks*

Eth hdr	Label hdr	Label hdr	IP hdr	IP payload
	stack=0	stack = 1		

Forwarding Equivalence Class (FEC)

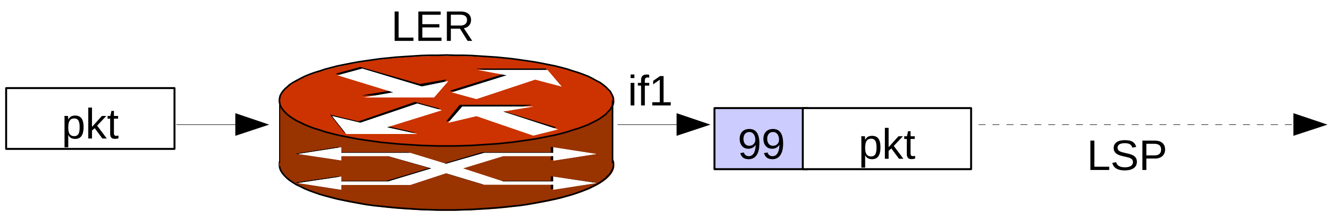
- Sort packets into different classes – classify them.
  - Classification is a more general form of lookup
  - Example1: All packets to one destination: ipdst = 192.168.20.33
  - Example2: All UDP packets with the ToS field set to 0x42 from sub-network 192.168.20.0/24
- Such a subset is called a Forwarding Equivalence Class (FEC)
- MPLS binds labels to FECs – Labelling
- FECs granularity
  - Coarse – good for scalability
  - Fine – good for flexibility
- What defines FECs?
  - “Something else”: eg BGP routing table, VPN, packet filters, etc.
- I.e., The meaning of a FEC (the semantics) is added by the overlying application

FEC	label
FEC1	99
FEC2	2345
...	...

Label operations

- Push a label (typically at ingress)
  - Double push (label stacking)
- Swap a label
  - Made by internal LSRs / P routers
- Pop a label
  - Typically at egress or pen-ultimate LSR
- Label operations are interface-specific
  - Since labels are unique between LSRs

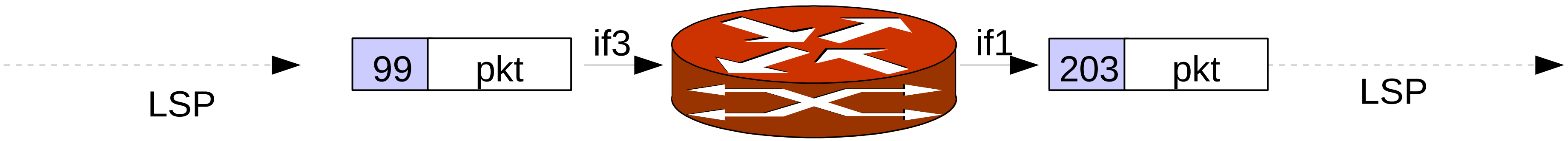
Label pushing



FEC	op	out-label	interface
F1	push	99	if1

- The border router (LER) classifies packets into FECs
- Binds a label to the packet
  - –Actually it maps the FEC to an LSP which in turn defines a label
- Pushes an MPLS header on the packet
- And sends it on the outgoing interface of the LSP

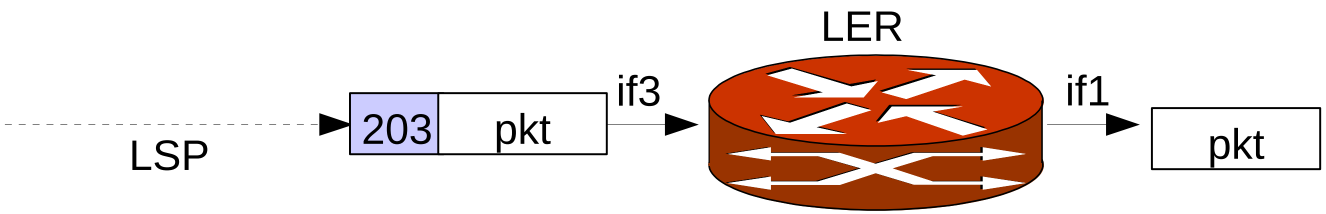
Label swapping



In-Interface	in-label	op	out-label	out-interface
if 3	99	swap	203	if1

- A router (LSR) makes a label lookup and swaps the label
- Rewrites the MPLS header
- And sends it further on the LSP

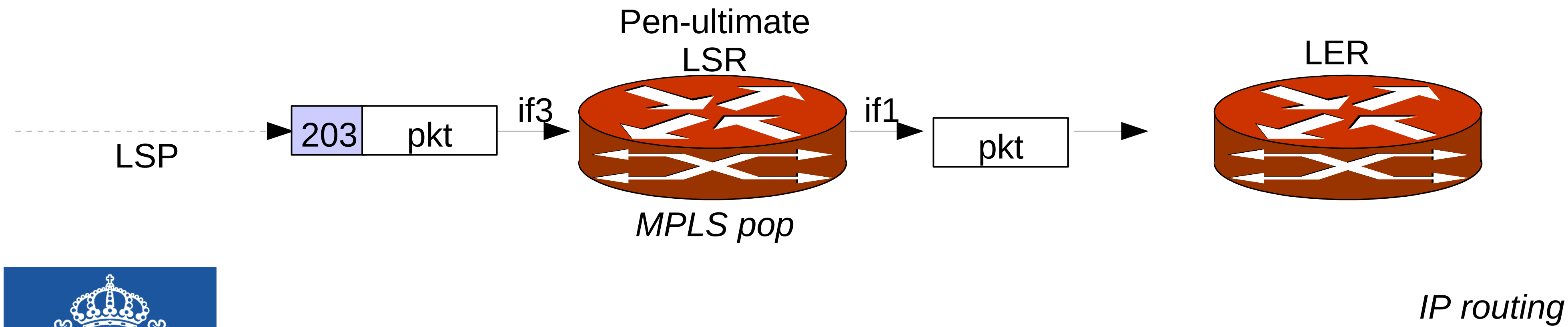
Label popping



In-Interface	in-label	op	out-label	out-interface
if 3	203	pop		(IP lookup)

- The border router (LER) pops the MPLS packet
- And then forwards it as usual depending on the packets protocol
- Example: pkt is an IP packet --> pkt is sent to IP forwarding
  - Thus both MPLS and IP forwarding on same node!

Pen-ultimate popping



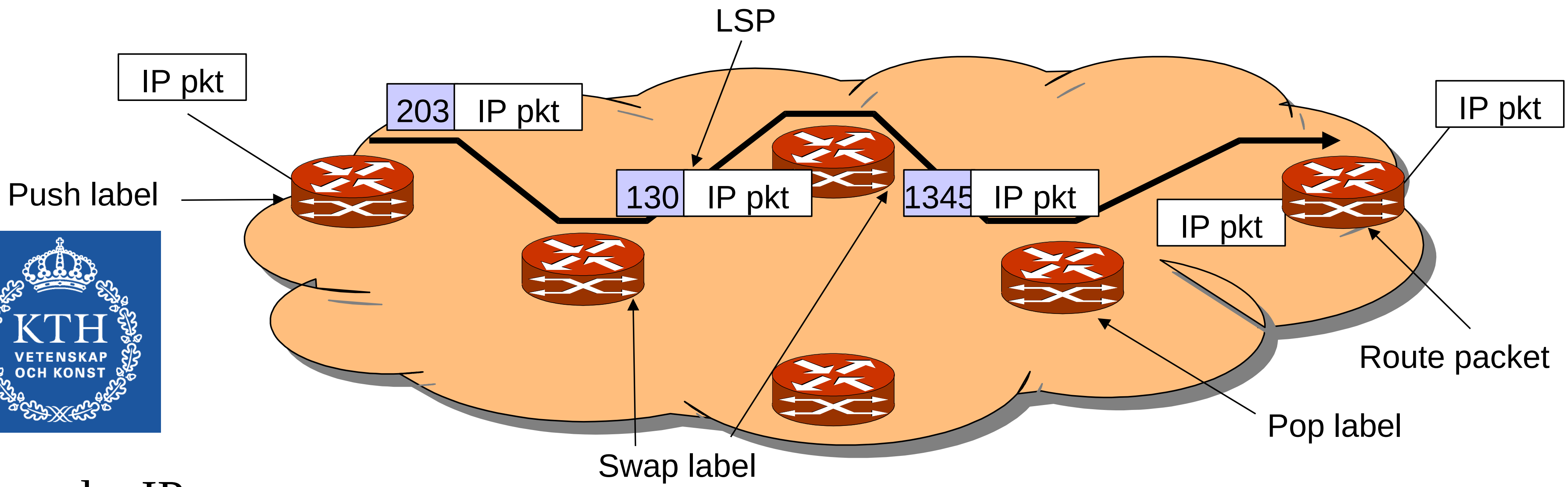
In-Interface	in-label	op	out-label	out-interface
if 3	203	pop	-	if1

- To make it easier for the border router, pop the label on the previous router (pen-ultimate)
- The pen-ultimate LSR does MPLS pop
- The LER does only IP routing

Special labels operations

- 0: IPv4 explicit NULL
  - –Downstream LSR should pop label unconditionally
  - –Popped packet is an IPv4 datagram
- 1: Router alert
  - –Deliver to control plane – do not forward
- 2: IPv6 explicit-NULL
  - –Downstream LSR should pop unconditionally
  - –Popped packet is an IPv6 datagram
- 3: Implicit-NULL
  - –Pop immediately and treat as IPv4 packet
  - –Note: This label does not actually appear on link (virtual)
  - –Use for pen-ultimate popping!

MPLS Label Switched Paths



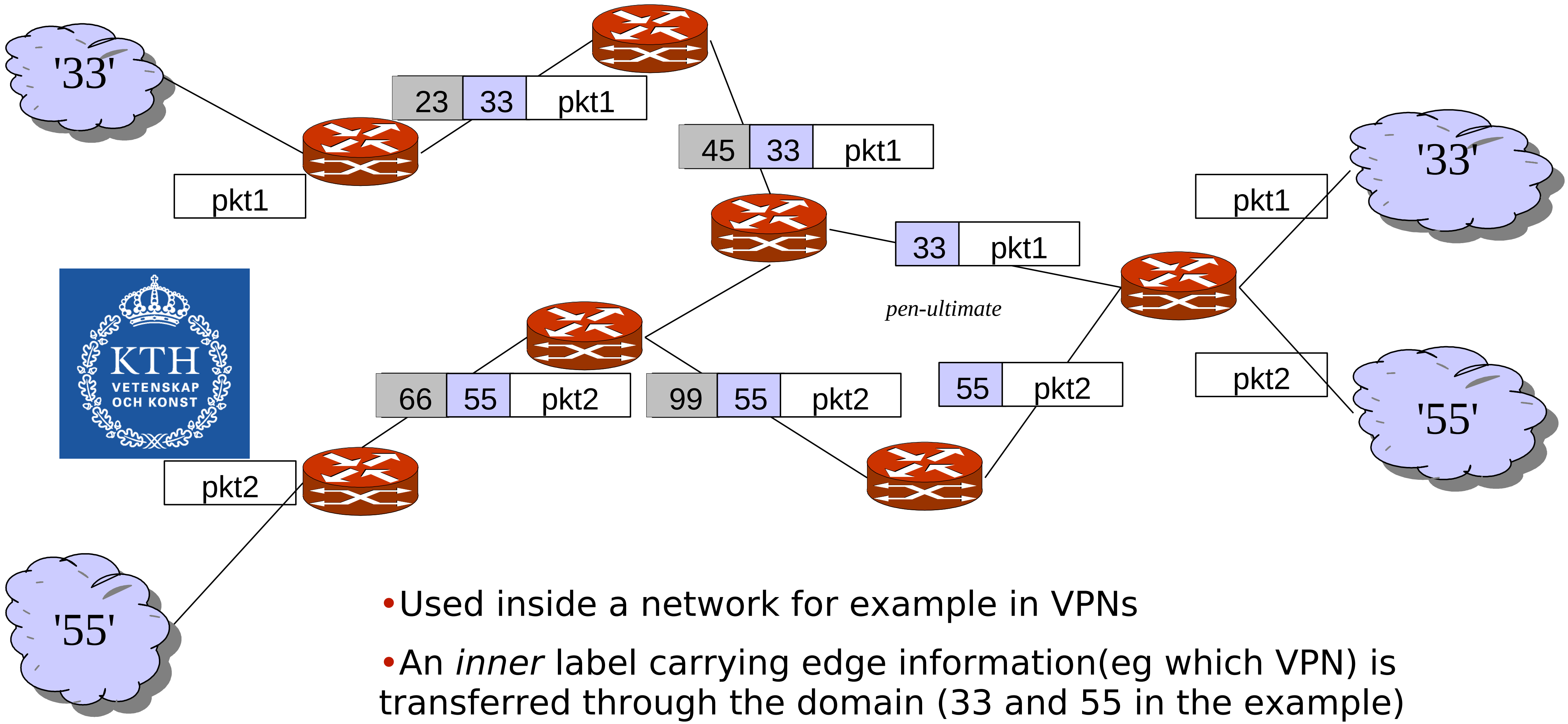
Example: IP

Pen-ultimate popping

Eth hdr	Label hdr	IP hdr	IP payload

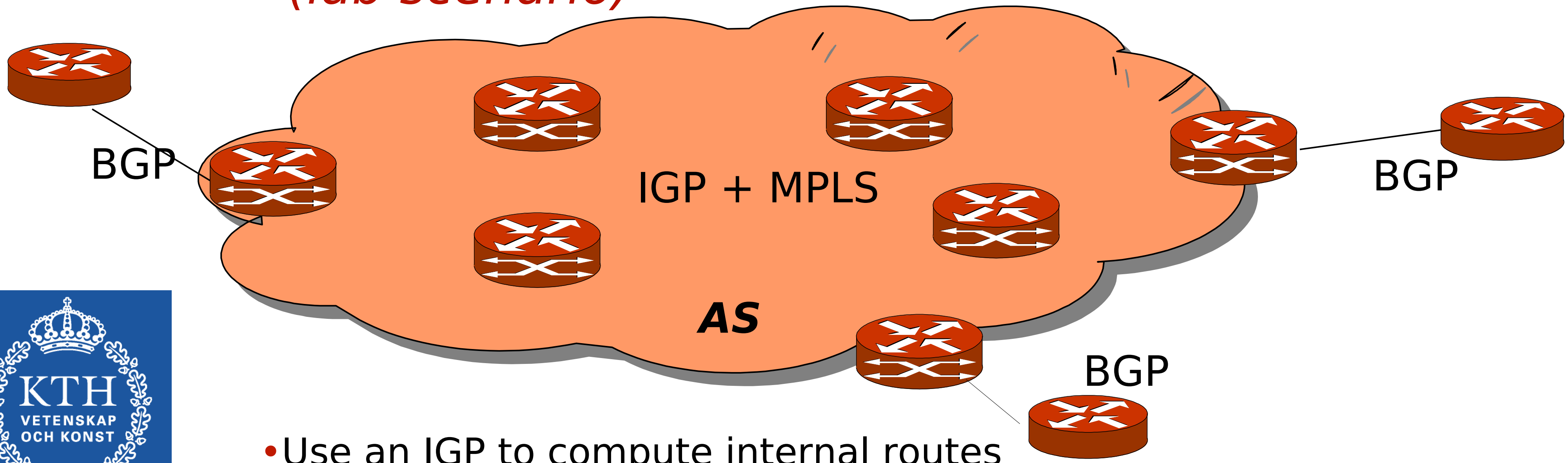
20-bit label	3-bit exp	1-bit stack	8-bit TTL

Label Stacking



- Used inside a network for example in VPNs
- An *inner* label carrying edge information(eg which VPN) is transferred through the domain (33 and 55 in the example)
- *Outer* label is used for transfer through the network
- Push both labels at ingress.
- Pop outer label at pen-ultimate LSR

Typical use: MPLS for *transit* *(lab scenario)*



- Use an IGP to compute internal routes
- Setup LSPs between border routers using the IGP
  - Eg border routers may set up a *full-mesh* of LSPs
- Send *transit traffic* via LSPs (src and dst outside the AS)
- But still send *internal* traffic via IP (src or dst inside the AS)
- External routes need not be distributed to non-border routers, so we do not need IBGP there
  - A *BGP-free core*
- Only the border routers need to speak BGP
  - This is considered an advantage

MPLS for transit

- Having MPLS in the core thus means that you do not need IBGP in interal routers
  - You still (always) need IBGP between border routers
- Thus, MPLS is an alternative to using IGP internally for external routes (bad choice) or IBGP (suffers from scaling)
- Note that interbal traffic still uses IP, only transit goes in MPLS
- We need some tricks in the border routers to make transit traffic follow LSPs and not IP
  - separation between transit and internal traffic
  - In JunOS this is done with inet.3 for next-hop routes (last slide)

Label Distribution

- Labels need to be assigned and LFIBs programmed
- A signaling protocol distributes labels
  - –Creates an LSP through an MPLS network
- There are different ways to do this
- 1. Make a new protocol
  - –LDP – Label Distribution Protocol
- 2. Extend existing protocols
  - –BGP – Border Gateway Protocol
  - –RSVP – Resource Reservation Protocol
- These protocols all distributes labels
  - –But they are somewhat different and can be combined to transfer different labels, eg BGP+RSVP, where BGP transfers *inner* labels and RSVP negotiate *outer* labels.
  - –RSVP is great for bandwidth reservation,...

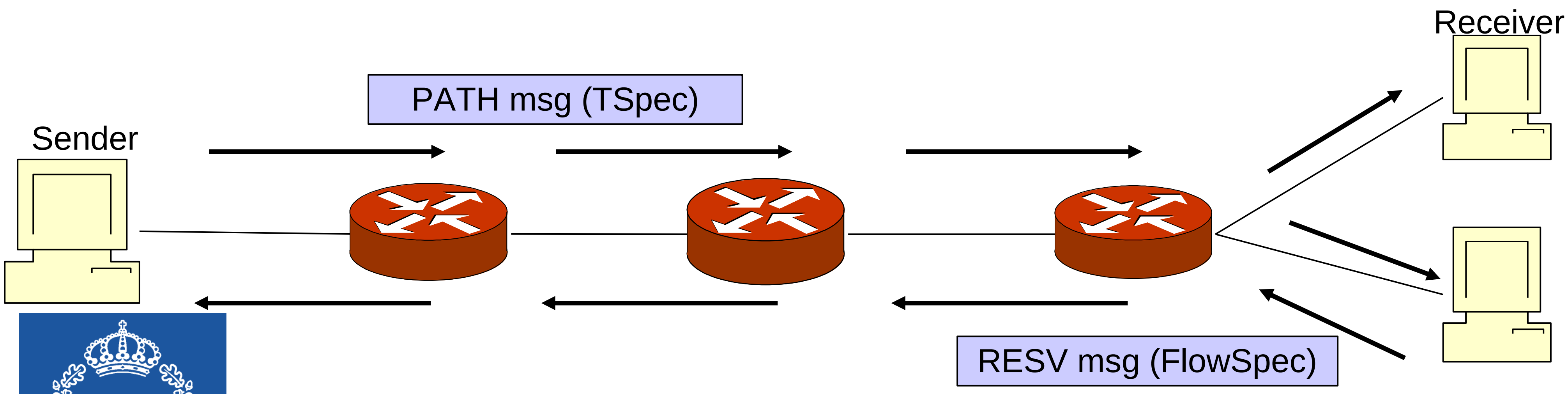
Label distribution and IGP

- Label distribution protocols typically rely on an IGP (eg OSPF/IS-IS) to find the shortest path of their LSP.
  - Or use source-routing to do constrained SPF or fixed-path routing
- They then assign labels to the LSP on the path.
- Normally, labels are assigned in *upstream* direction
- During setup of LSPs, traffic may be *black-holed*
  - Discrepancy between IGP and label distribution
  - Same is true during reconvergence (IGP routes change)

Resource Reservation Protocol

- RSVP is a network control protocol used to express quality of service (QoS).
  - Binds a QoS request to a flow
- RSVP delivers QoS reservations along a path from source to destination(s).
  - No routing: IGP computes path
  - Uses ”soft-state”: paths are recomputed when routing changes
- RSVP-TE is used for traffic engineering for MPLS

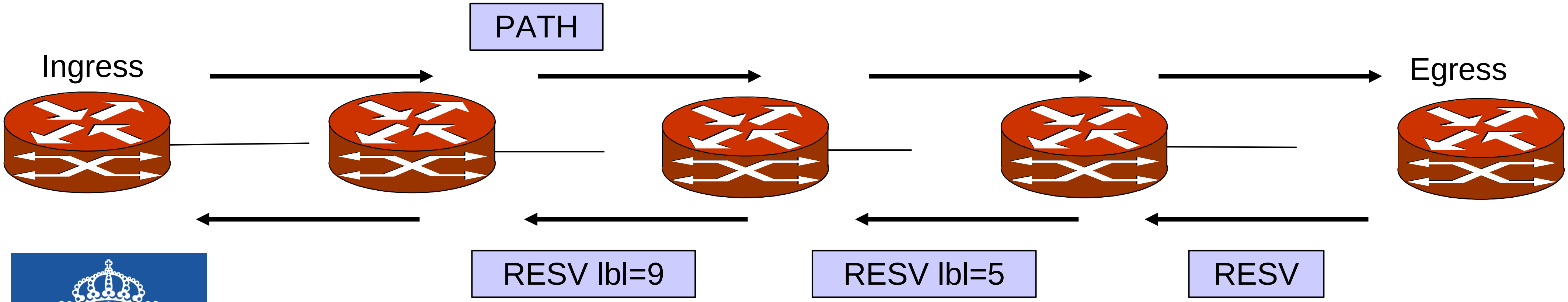
RSVP Model



Two types of messages to set up reservations:

- PATH message
  - From a sender to one or several receivers, carrying TSpec and classification information provided by sender
- RESV message
  - From receiver, carrying FlowSpec indicating QoS required by receiver

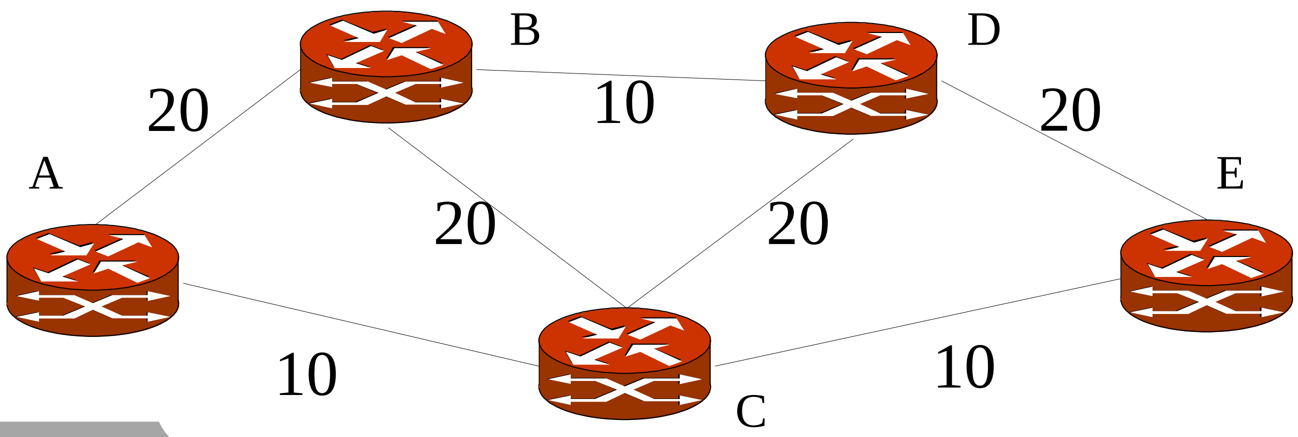
MPLS support in RSVP-TE



- RSVP-TE has been adapted to support MPLS
- New objects defined carrying labels
- In this way, RSVP-TE can express QoS (Tspec, FlowSpec) associated with an LSP
  - Network resources are then bound to that LSP throughout the network.

Traffic engineering with RSVP-TE/MPLS

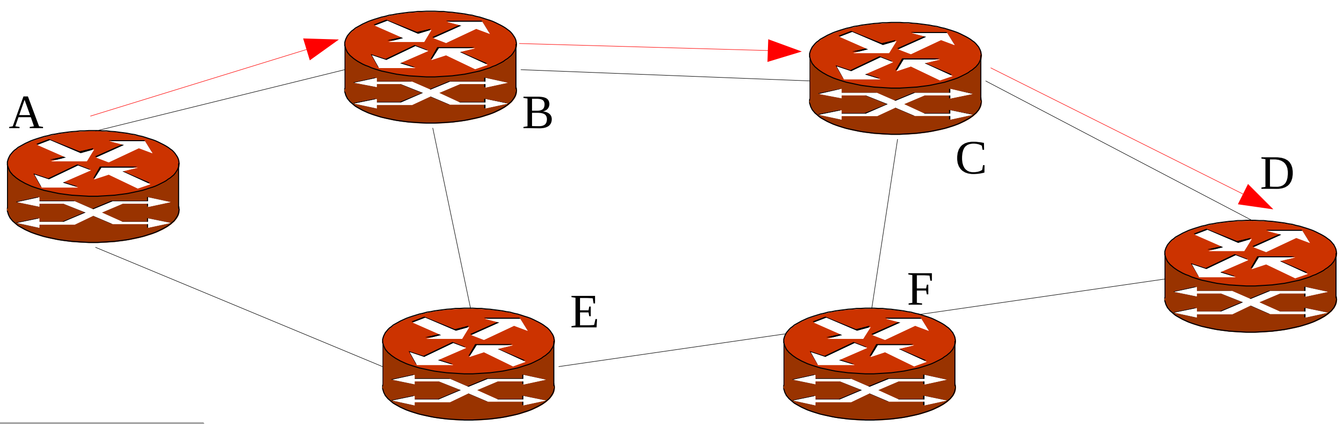
- RSVP can reserve resources in the network
- RSVP can signal alternative paths using
  - Constrained shortest path (eg using allocated bandwidth)
  - Strict/ loose source routing
- Why? Traffic distribution, alternate paths
  - Difficult to do with regular routing protocols.
- Example:
  - LSP strict source routing: A,B,C,D,E (20 Mbps)
- See routing algorithms lecture: CSPF



Protection switching in MPLS

- Assume a primary LSP is signalled from A-D via B and C
- If a link or node goes down, how is reliability ensured?
- There are several issues and techniques:
    - Detection of failure
    - IGP re-route
    - Path protection
    - Local protection

To think about
- Switchover latency
- Over-reservation

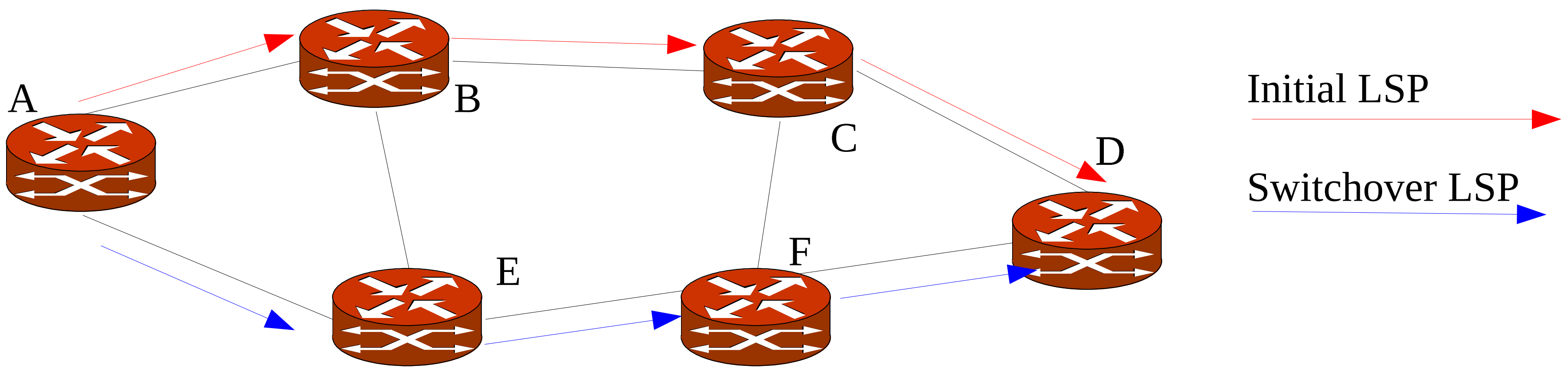


Detection of failure

- SDH/SONET does protection switching at ~50ms.
- Routing protocol Hello's are typically on 1s-10s.
- Physical level detection.
  - But not all links support this.
- Instead, send packets very often
- MPLS pings along the LSP
  - Send many packets and detect losses
- BFD - Bidirectional forwarding detection
  - Send many packets and detect losses
  - Generic technique for other protocols: OSPF, BGP, etc
- But sending many packets per LSP has its cost in bandwidth use
  - And CPU usage if not done in hw

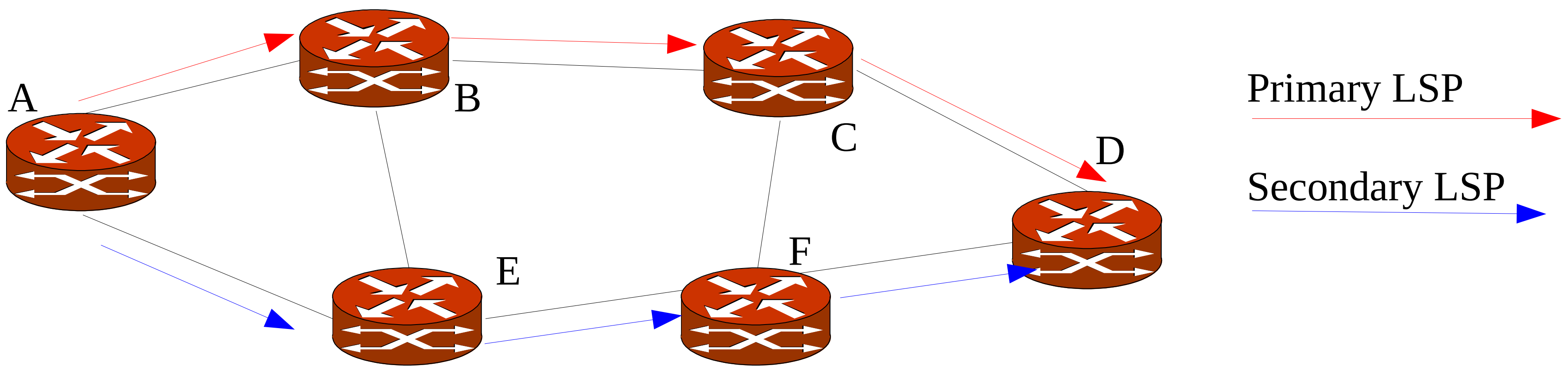
IGP reroute

- When the underlying IGP detects a failure, it will reroute around the failure, and thus RSVP-TE will send its PATH and RESV messages on the new route and the LSP will eventually establish itself using the new route
- The cleanest solution but may be slow if IGP protection switching and RSVP failover is slow.
- This is done in the lab



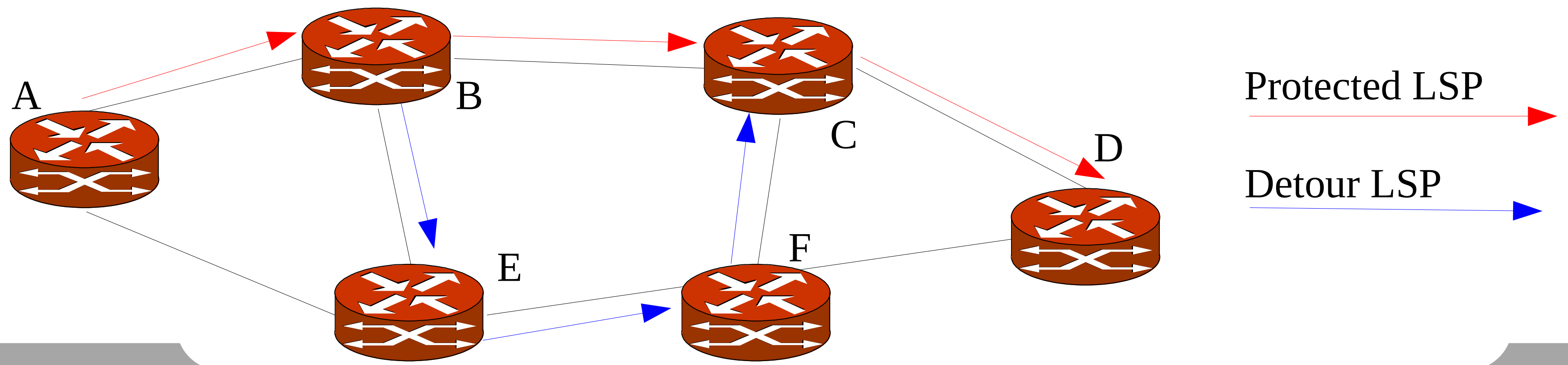
Path protection

- Compute a secondary path in advance
- Switchover when the primary path fails using BFD/MPLS pings
- This is also done in the lab



Local protection (FRR)

- Protected LSP: ready-made detours
- The repair is made locally by pre-computed detour LSPs
  - Fast switchover since reroute made locally
- A detour is an extra LSP from a node in the path to a merge point
  - Link or node protection
  - One-to-any or ”facility” LSP
- Example (link protection)
  - Protected LSP: A->B->C->D
  - Detours: A->E->B; B->E->F->C; C->F->D (only 2nd shown in fig)

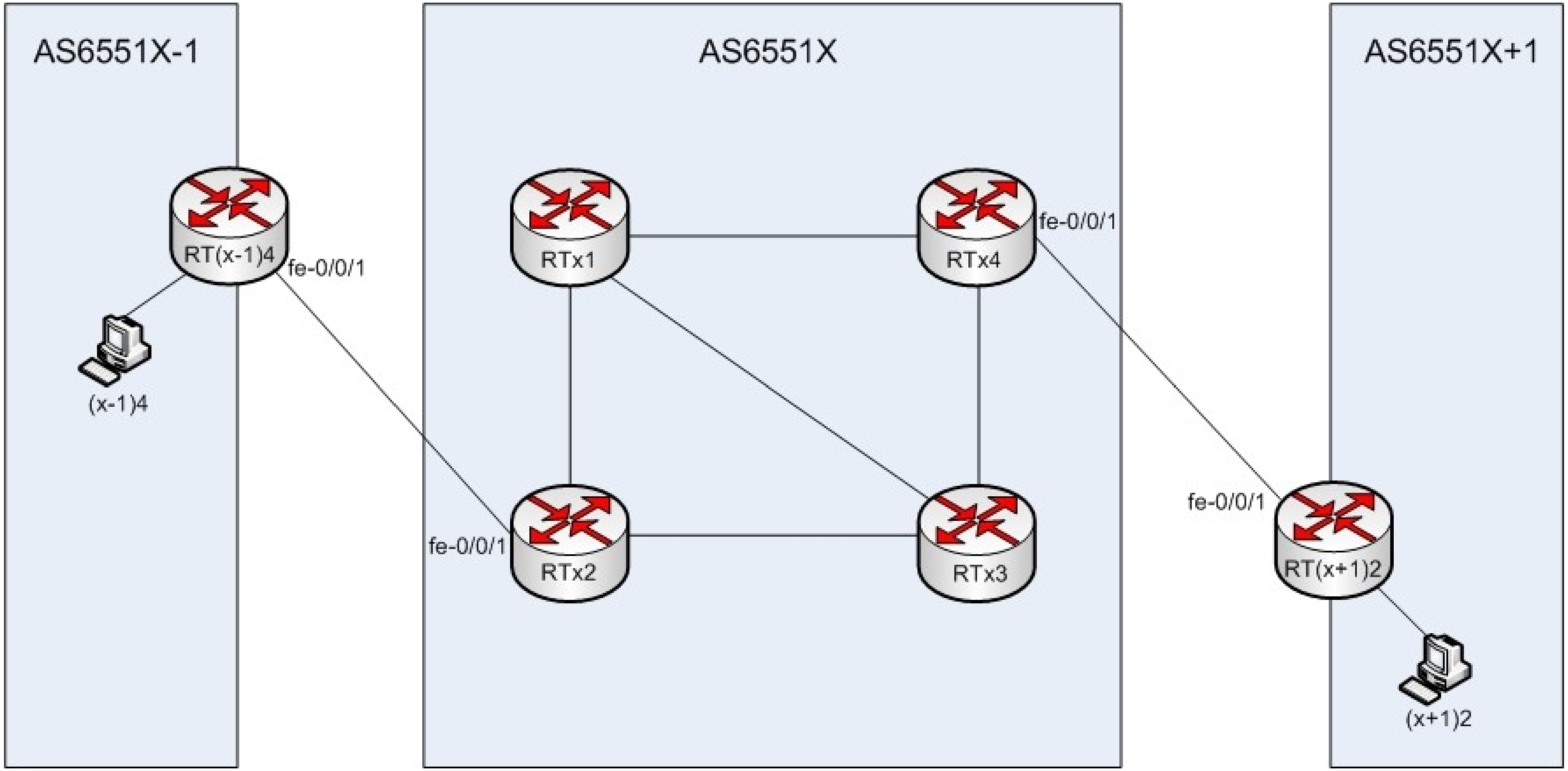


Summary: MPLS in a transit network

Putting it all together:

- IGP is used to routes between all routers in the network
- MPLS is used to carry transit traffic between border routers (PE)
- RSVP-TE is used to signal MPLS LSPs between border routers.
- RSVP-TE is used to reserve bandwidth in LSPs
- RSVP-TE used to compute alternative paths for switchover
- BGP is used for all external routes.
- This is the lab scenario.

Lab: Using MPLS for transit: BGP-free core



MPLS in JunOS

- See http://www.juniper.net/techpubs/software/junos/junos94/swconfig-mpls-apps
- Example:

  Enable mpls on all forwarding interfaces

  Enable icmp in mpls for debugging (traceroute)

  Setup LSPs (using explicit path setup: no cspf)

```
interface so-0/0/0 {
    unit 0 {
        family mpls;  # Enable mpls address family
    }
}
protocols mpls {
    icmp-tunneling;             # Enable icmp for debugging
    interface so-0/0/0.0;       # Include interface in mpls forwarding
    label-switched-path btoc {  # Define an LSP
         to 193.10.255.6;       # LSP end-point
         no-cspf;              # Enable explicit-path computation
    }
    rsvp {
        interface so-0/0/0.0; # Enable rsvp on interface
}
```

- cspf - Constrained Path Shortest Path

  Dont use it in lab - use explicit routing.

MPLS show commands

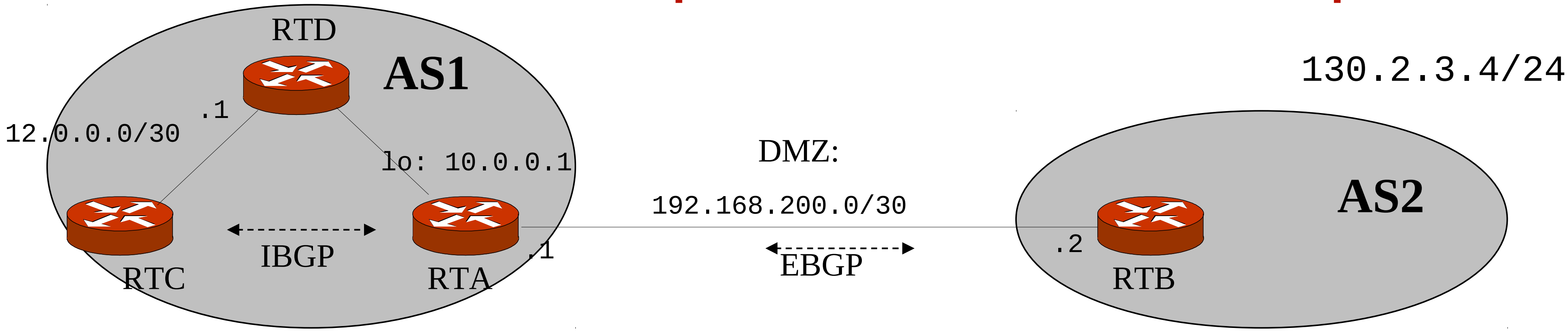
```
> show mpls lsp
Ingress LSP: 1 sessions
To              From            State Rt P     ActivePath       LSPname
193.10.255.6    193.10.255.5    Up     1 *                      btoc

Egress LSP: 1 sessions
To              From            State    Rt Style Labelin Labelout LSPname
193.10.255.5    193.10.255.6    Up        0  1 FF       3        - ctob

Transit LSP: 2 sessions
To              From            State    Rt Style Labelin Labelout LSPname
193.10.255.5    193.10.255.6    Up        1  1 FF  299792        3 ctob
193.10.255.6    193.10.255.5    Up        1  1 FF  299776        3 btoc

> show route protocol rsvp
299776              *[RSVP/7] 3d 18:23:44, metric 1
                     > via so-0/1/0.0, label-switched-path btoc
299776(S=0)         *[RSVP/7] 3d 18:23:44, metric 1
                     > via so-0/1/0.0, label-switched-path btoc
299792              *[RSVP/7] 3d 18:23:37, metric 1
                     > via so-0/0/0.0, label-switched-path ctob
299792(S=0)         *[RSVP/7] 3d 18:23:37, metric 1
                     > via so-0/0/0.0, label-switched-path ctob
> show mpls interface
> show rsvp interface
> show rsvp neigbour
> ...
```

EBGP nexthop: recursive lookup



RTC:s routing table alternatives:

Next-hop self is necessary for BGP to use MPLS!

Route	Nexthop	Protocol
130.2.3.4/24	192.168.200.2	IBGP
192.168.200.0/30	12.0.0.1	IGP *DMZ nexthop*
10.0.0.1/32	lspA	RSVP
12.0.0.0/30	-	direct

Route	Nexthop	Protocol
130.2.3.4/24	*10.0.0.1*	IBGP
10.0.0.1/32	12.0.0.1	IGP
10.0.0.1/32	lspA	RSVP *Next-hop self*
12.0.0.0/30	-	direct

Next-hop self issue

- There are routes both from IGP and RSVP. Internal traffic should use the IGP, external should use RSVP.
  - RSVP adds end-points in a separate inet table which BGP uses: inet.3. RSVP has lower precedence than the IGP
  - BGP looks in both inet.0 and inet.3. IGP does not. This is how transit traffic uses the LSP tunnels.