
cisco Live!
ALL IN

The bridge to possible

SR IGP Flex-Algo

Kamrul Islam – Customer Delivery Architect, @kamrul2525
Matt Breneisen – Customer Delivery Architect, @mattbreneisen

BRKMPL-2129

Cisco Webex App

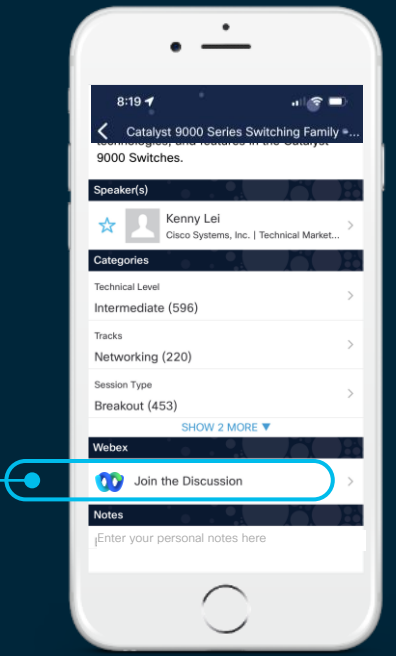
Questions?

Use Cisco Webex App to chat
with the speaker after the session

How

1. Find this session in the Cisco Live Mobile App
2. Click “Join the Discussion”
3. Install the Webex App or go directly to the Webex space
4. Enter messages/questions in the Webex space

Webex spaces will be moderated by the speaker until June 17, 2022.



https://ciscolive.ciscoevents.com/ciscolivebot/#BRKMPL-2129

Why are we here?

- Wouldn’t it be nice to be able to manipulate the IGP to our own specific needs? If we could add our own attribute or constraint?
  - Only use a route with a cumulative delay based on a measured per link delay
  - Have a highly reliable network achieving 5 or even 6 9’s of uptime.
  - If we only want to use a secure path. Such as paths with MACsec only
  - Define a path traversing high speed links for bandwidth sensitive traffic
  - Only use a subset of the routers in your network

Agenda

- MPLS Traffic Engineering Evolution
- SR IGP Flexible-Algorithm (Flex-Algo)
- SRTE ODN Policy using Flex-Algo
- Use Cases
- Conclusion

What we won't be able to cover

- Basics of Segment Routing
- ISIS or OSPF basics of label transport

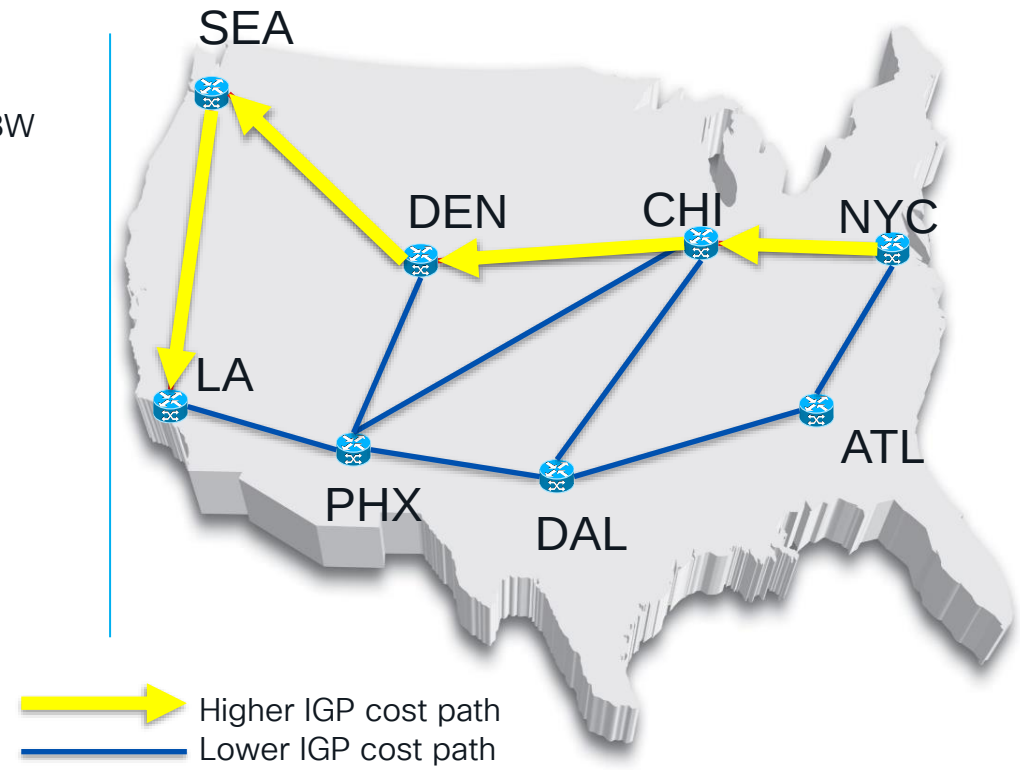
Recommended Sessions to review or attend

- BRKMPL-2137 Designing MPLS based IP VPNs
- BKRMPL-2131 Deploying VPNs Over Segment Routed Networks Made Easy
- BRKMPL-2117 SRv6 based IP Transport-Design, Deployment Best Practices & Challenge
- BRKMPL-2123 Multicast Segment Routing & Traffic Engineering
- BRKMPL-2119 Traffic-Engineering with SR and SRv6 Evolution

MPLS Traffic Engineering Evolution

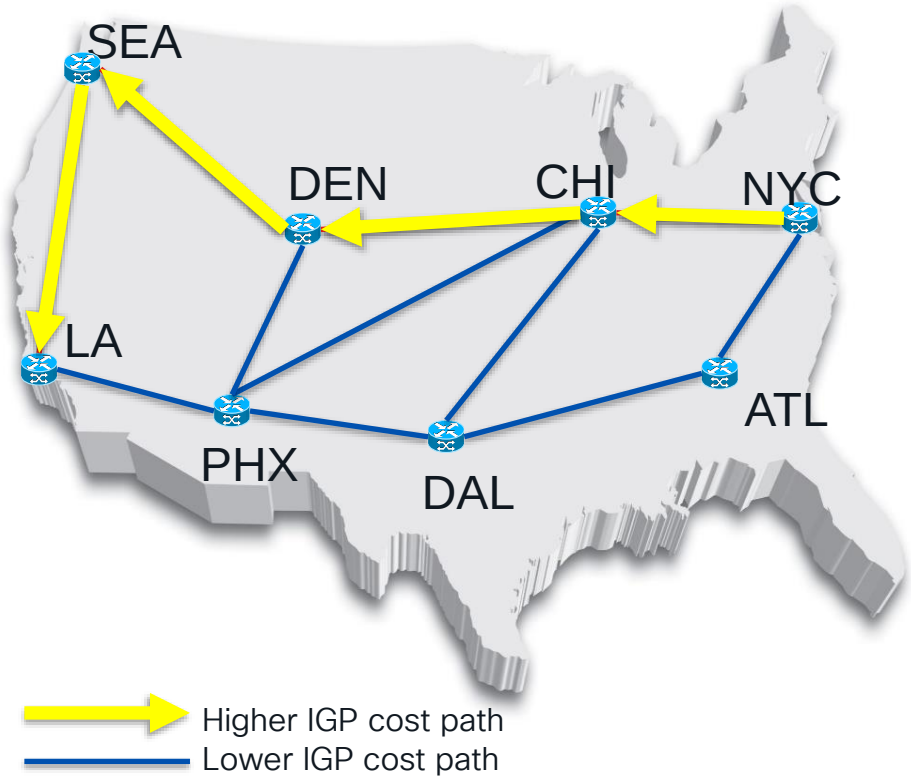
MPLS TE with RSVP

- In addition to MPLS and IGP
  - RSVP use for path signaling, label distribution, BW control, etc.
- TE Path
  - TE tunnel from NYC to LA via DEN and SEA
- Path computation is complex & lack of scalability
  - RSVP control plane is complex
  - To protect the primary TE path, required FRR backup tunnels for link/node protection
  - TE states build in all the nodes from head-end tail-end
  - RSVP-TE is not ECMP friendly



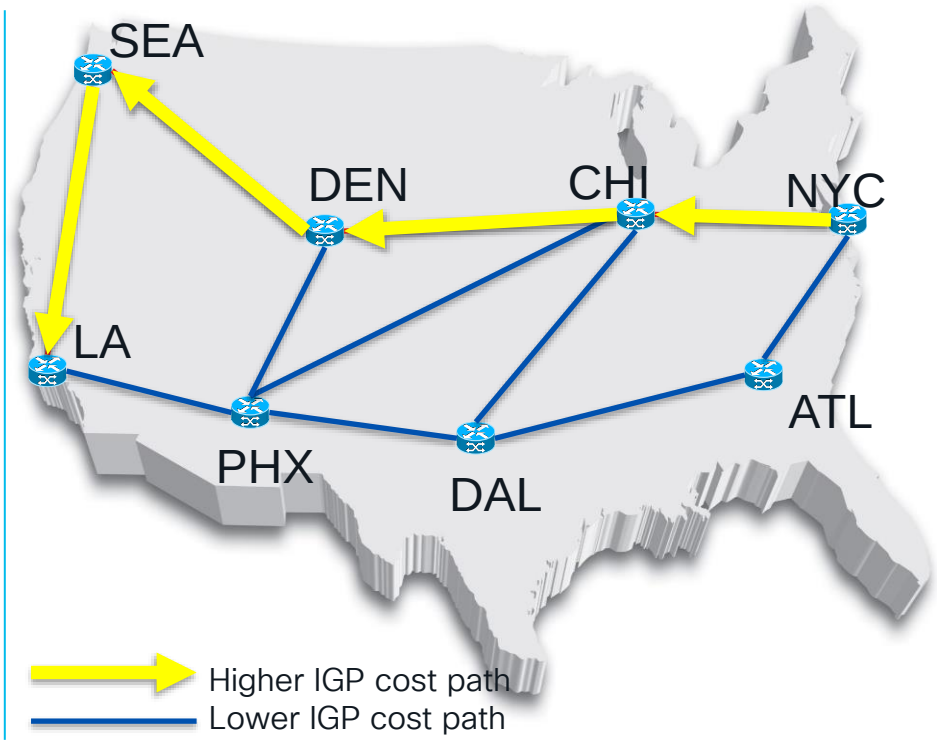
MPLS TE with Segment-Routing / SR-TE Policy

- No additional Protocol i.e., No RSVP except IGP and MPLS
  - Source-based routing
  - SR Labels (SID) are distributed by IGP
- TE Path
  - TE tunnel from NYC to LA via DEN and SEA
- Simplify and Scale Better
  - Head-end accumulates labels to reach destination, align with centralized controller concept
  - TE states build only at head-end node
  - TI-LFA natively support sub 50ms link/node protection and uLoop avoidance
  - Inherent support of ECMP and UCMP



SR ODN Policy using Flex-Algo

- In addition to SR-TE benefits, Flex-Algo brings the following added capabilities:
  - On-Demand Next-hop (ODN) Policy
  - Automated Steering (AS) based on intent
  - Support Inter-domain latency and SRLG for disjoint path
  - Use Flex-algo label, no Adj label i.e., decrease number of labels in packet header



SR IGP Flex-Algo

SR IGP Flex-Algo (Flexible-Algorithm)

- Flex-Algo is a mechanism that allows a network operator to influence a path computation by associating Metric and Constraints to Flex-Algo instead of using link-cost based SFP
- Flex-Algo instance (K) is defined as
  - Metric: IGP or Latency or TE
  - Constraints: Exclude/Include Link-affinity (Link color), Shared Risk Link Group (SRLG) for path dis-jointness,
- TI-LFA honors Flex-algo constraints for backup path
- Flex-algo is distributed by ISIS/OSPF

Currently Defined Flex-Algo(s)

- Flex-Algo(s): 0-255, 0-127 are reserved, 128-255 are Operator Useable

- Algo 0: Shortest Path First (SPF) algorithm based on IGP metric

  This is the well-known shortest path algorithm as computed by the IS-IS decision process. consistent with the deployed practice for link-state protocols, algorithm 0 permits any node to overwrite the SPF path with a different path based on local policy

- Algo 1: Strict Shortest Path First (SPF) algorithm based on IGP metric

  The algorithm is identical to algorithm 0 but algorithm 1 requires that all nodes along the path will honor the SPF routing decision. Local policy MUST NOT alter the forwarding decision computed by algorithm 1 at the node claiming to support algorithm

Flex-Algo Prefix-SID

- Flex-Algo Participation Advertisement - node wants to participate in Flex-algo
- No additional loopback address is needed

Node belongs to Flex-algo(FA):

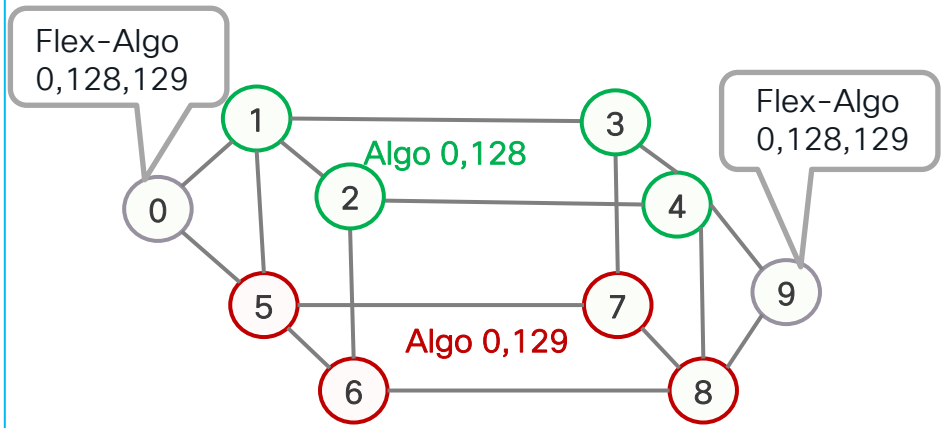
Nodes 0 and 9 participate to Algo 0, 128 and 129
Nodes 1,2,3 and 4 participate to Algo 0 and 128
Nodes 5,6,7 and 8 participate to Algo 0 and 129

Node 3 FA Participation Adv:

Prefix-SID for FA 0 = loopback0 +FA 0 + 16300
Prefix-SID for FA 128 = loopback0 +FA 128 + 16308

Node 9 FA Participation Adv:

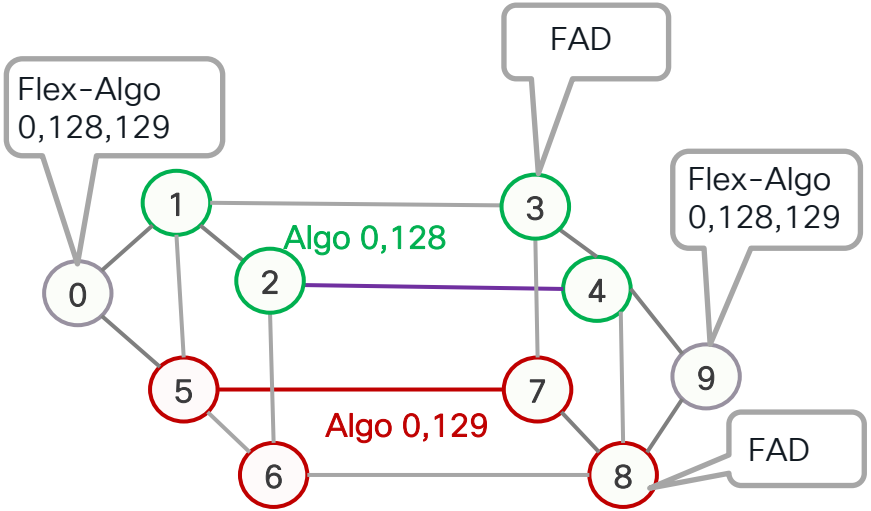
Prefix-SID for FA 0 = loopback0 + FA0 +16900
Prefix-SID for FA 128 = loopback0 + FA128 +16908
Prefix-SID for FA 129 = loopback0 + FA129 +16909



Flex-algo(s) - 128-255, operator useable
Default SRGB is 16000 - 23999

Flex-Algo Definition (FAD)

- Operator can associate the desired metric type, and constraints to Flex-algo(s)
  - IGP metric or latency metric or TE metric
  - Link affinity and/or SRLG
- All nodes MUST agree on same definition of the Flex-Algo(s) for loop free forwarding
  - Example, Node 3 and 8 Advertisement:
  - FA 128 = metric is IGP + exclude purple link
  - FA 129 = metric is delay + exclude red link
- Multiple FAD nodes are recommended for redundancy



Flex-Algo Computation and Install Prefix-SID

- Let's say node N needs to compute a path using Flex-Algo 128
  - Node N is needed to enable Flex-algo 128 for participation
  - Node N has a consistent definition for algo 128
  - Node N supports the definition for algo 128

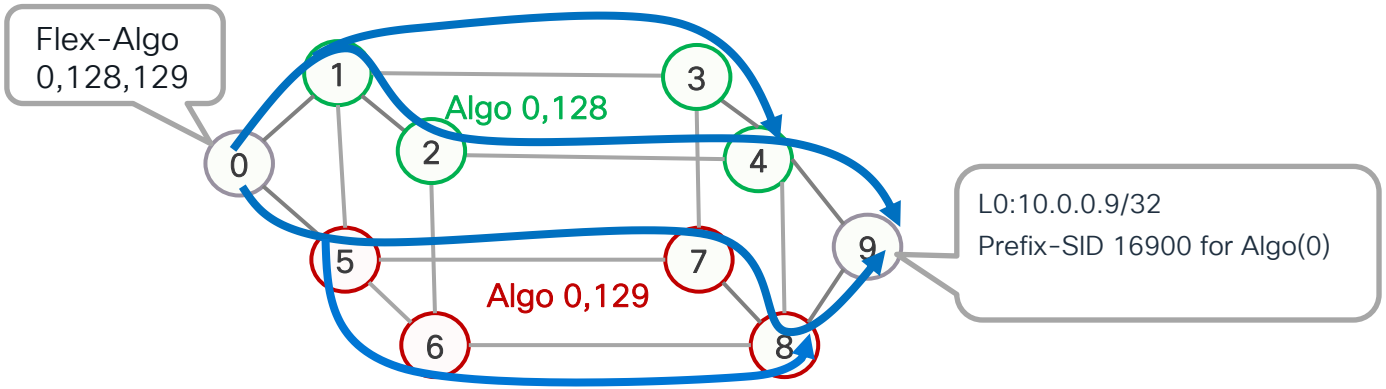
- 1st step is to define the topology for algo 128
  - Node N prunes any node that is not advertising participation to algo 128
  - Node N prunes any link that is excluded by the algorithm of algo 128

    e.g., if 128 excludes link-affinity RED then any link with link-affinity RED is pruned

Flex-Algo Computation and Prefix-SID installation

- 2nd step is to compute shortest-path tree for Topo(128) with the metric defined by 128
  - it could be the IGP metric, the TE metric or the delay
- 3rd step is to install reachability for Prefix-SID Flex-Algo 128 in the forwarding table

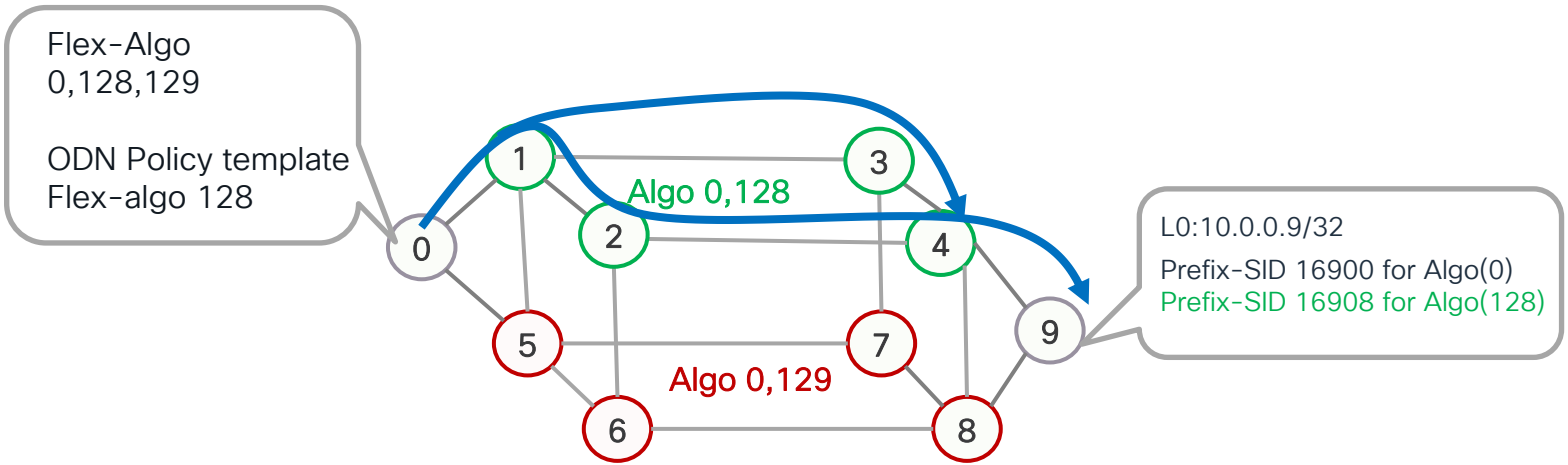
Example – Prefix-SID 16900 of Algo(0)



Let's say same IGP metric in all links, will follow ECMP

- Algo 0 is default flex-algo and metric type is IGP, will use entire topology
- SR leverage TI-LFA and uLoop

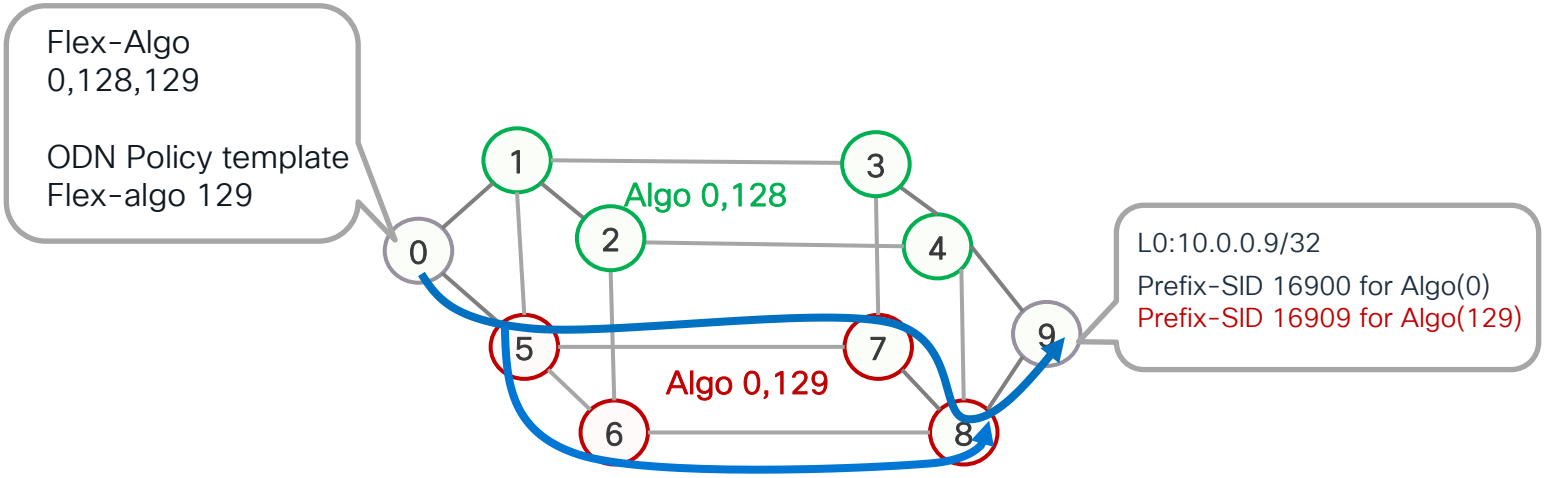
Example – Prefix-SID 16908 of Algo(128)



Let's say same IGP metric in all links, will follow ECMP

- Algo 128 is operator defined, compute path thru the nodes participate to algo 128
- TI-LFA backup path honor constraints and uLoop

Example – Prefix-SID 16909 of Algo(129)



Let's say same IGP metric in all links, will follow ECMP

- Algo 129 is operator defined, compute path thru nodes participate to algo 129
- TI-LFA backup path honor constraints and uLoop

SR ODN Policy using Flex-Algo

SR ODN Policy

- Per-Destination policy
  - Steer traffic based on next-hop and color of a BGP service route
  - Color is a BGP extended community attribute
  - Color is used for transport SLA indicator, for instance min-delay or min-cost
- Per-Flow policy
  - Steer traffic based on incoming packets classification (IPP, DSCP, ACL, EXP etc.)
  - Then set local Forward-Class up to 8, range 0-7
  - An ingress PBR policy applied to an input interface

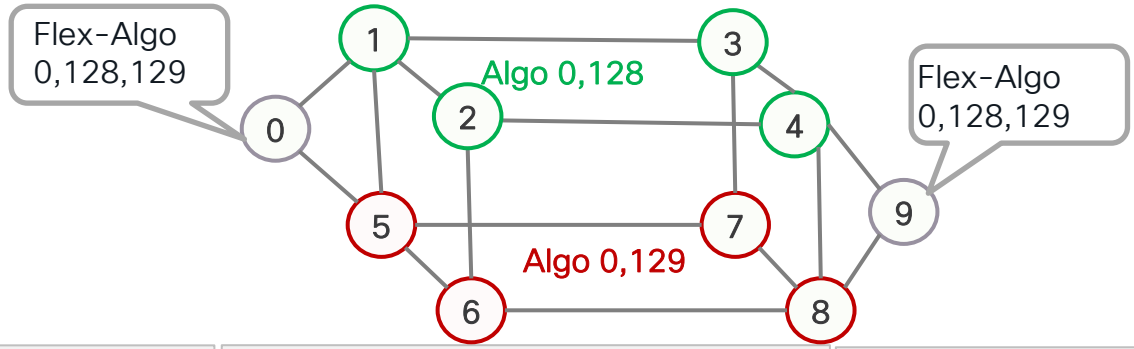
SR ODN Policy – Per Destination

- Steps to build SR ODN policy

1. Flex-Algo Prefix-SID, node wants to participate in Flex-algo
2. Flex-Algo Link Affinity-map, exclude or include a link for path computation by Flex-algo
3. Flex-Algo Definition (FAD), associate metric & link constraints to Flex-algo
4. BGP Extended Color Community, add color to BGP prefix for intended ODN path
5. Finally, SR ODN Policy, auto dynamic path computation based on intended SLA

1. Flex-Algo Prefix-SID

- Flex-Algo Prefix-SID is the node that wants to participate in Flex-algo(s)
- Use existing loopback address



Node3: IOS-XR

```
router isis 1
Flex-algo 128
interface Loopback0
 passive
 address-family ipv4 unicast
   prefix-sid absolute 16300
   prefix-sid algorithm 128 absolute 16308
```

Node7: IOS-XE

```
router isis 1
Flex-algo 129
!
segment-routing mpls
connected-prefix-sid-map
  address-family ipv4
   10.0.0.7/32 absolute 16700 range 1
  exit-address-family
  address-family ipv4 algorithm 129
   10.0.0.7/32 absolute 16709 range 1
  exit-address-family
```

Node9: IOS-XR

```
router isis 1
Flex-algo 128
Flex-algo 129
interface Loopback0
 passive
 address-family ipv4 unicast
   prefix-sid absolute 16900
   prefix-sid algorithm 128 absolute 16908
   prefix-sid algorithm 129 absolute 16909
```

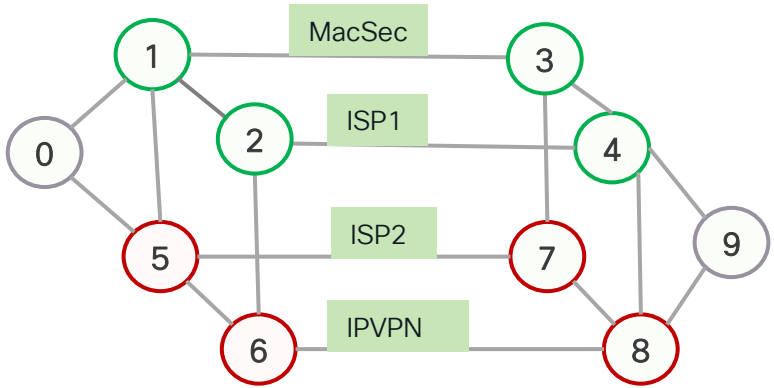
2. Flex-Algo Link affinity-map (Link color)

Node 1 # IOS-XR

```
Router isis 1
affinity-map MACSec bit-position 1
affinity-map ISP1 bit-position 2
affinity-map ISP2 bit-position 3
affinity-map IPVPN bit-position 4
!
interface tenG0/0/0/1
affinity flex-algo MACSec
```

Node 5 # IOS-XE

```
Router isis 1
affinity-map MACSec bit-position 1
affinity-map ISP1 bit-position 2
affinity-map ISP2 bit-position 3
affinity-map IPVPN bit-position 4
!
interface tenG0/0/0/1
Isis affinity flex-algo
name ISP2
```



Link affinity-map bit-position MUST be matched in all the nodes within the path computation domain.

3. Flex-Algo Definition (FAD)

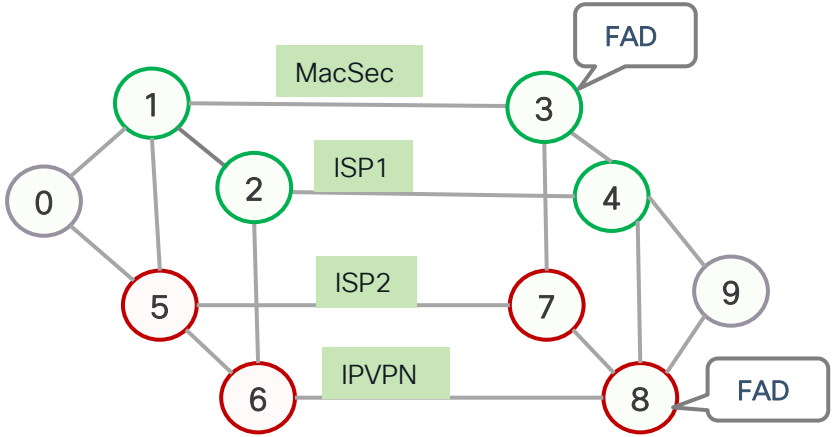
- ISIS uses Sub-TLV and OSPF uses TLV to advertise FAD

Node 3 # IOS-XR

```
router isis 1
flex-algo 128
priority 250
(By-default metric-type is IGP)
advertise-definition
affinity exclude-any ISP1
!
```

Node 8 # IOS-XE

```
router isis 1
flex-algo 129
  advertise-definition
  metric-type delay
  priority 250
```

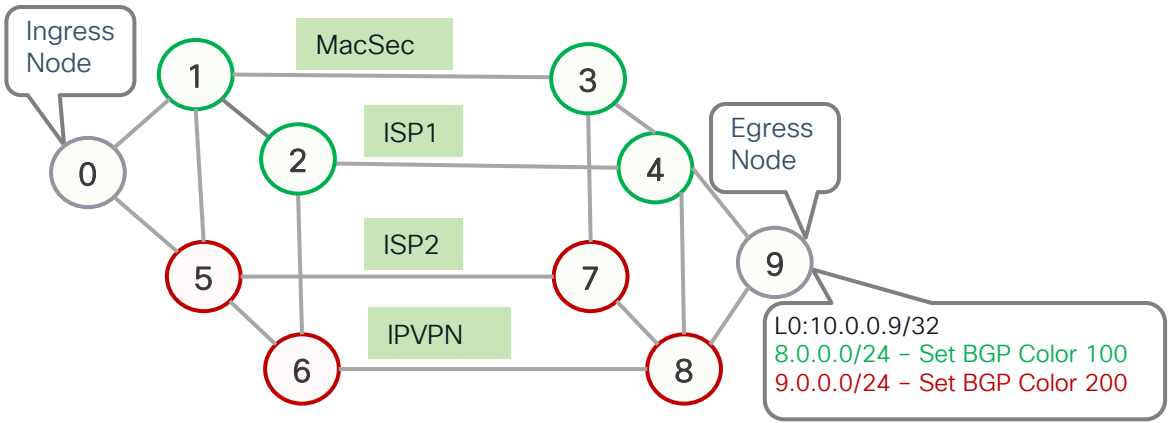


Select more than one node to advertise FAD with priority for active and standby FAD

4. BGP Extended Color Community

- Setup BGP extended color community for SLA

```
Node9 # IOS-XR
extcommunity-set opaque COLOR_100
  100
end-set
extcommunity-set opaque COLOR_200
  200
end-set
prefix-set PREFIX_8
  8.0.0.0/24
end-set
prefix-set PREFIX_9
  9.0.0.0/24
end-set
!
route-policy BGP_COLOR
  if destination in PREFIX_8 then
    set extcommunity color COLOR_100
  else
    if destination in PREFIX_9 then
      set extcommunity COLOR_200
    else
      pass
end-policy
```

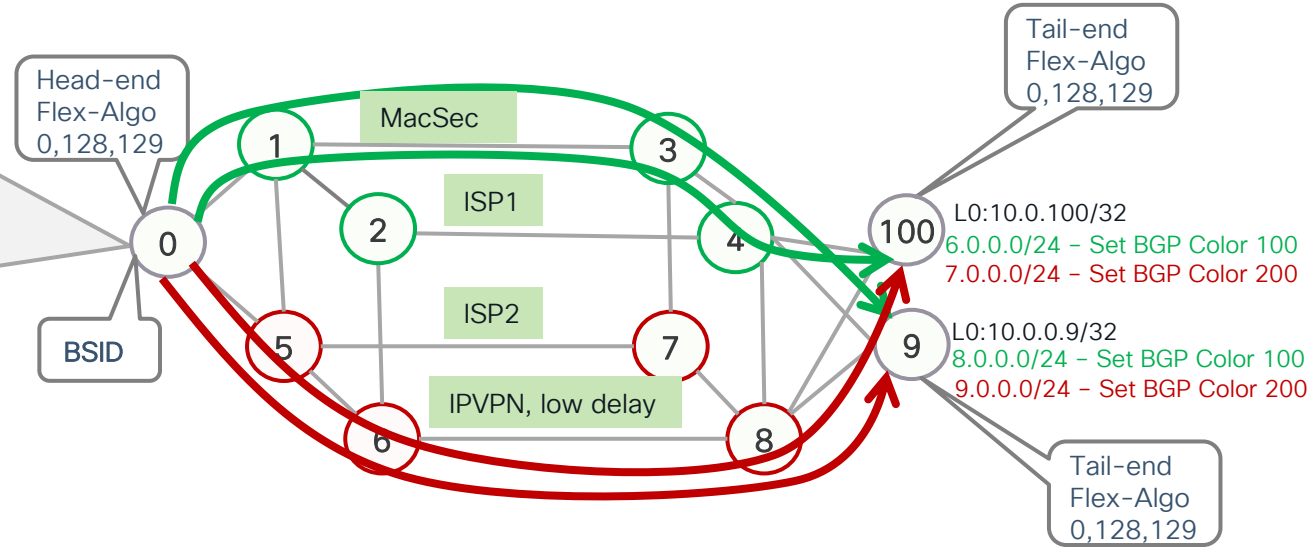


5.1 SR ODN Policy – Candidate Path

- Automated Steering by leveraging IGP Flex-Algo

Node0 # IOS-XR

```
segment-routing
 traffic-eng
!
 on-demand color 100
  steering
    path-invalidation drop
    restrict IP_ACL
   dynamic
    sid-algorithm 128
!
on-demand color 200
   dynamic
    sid-algorithm 129
```



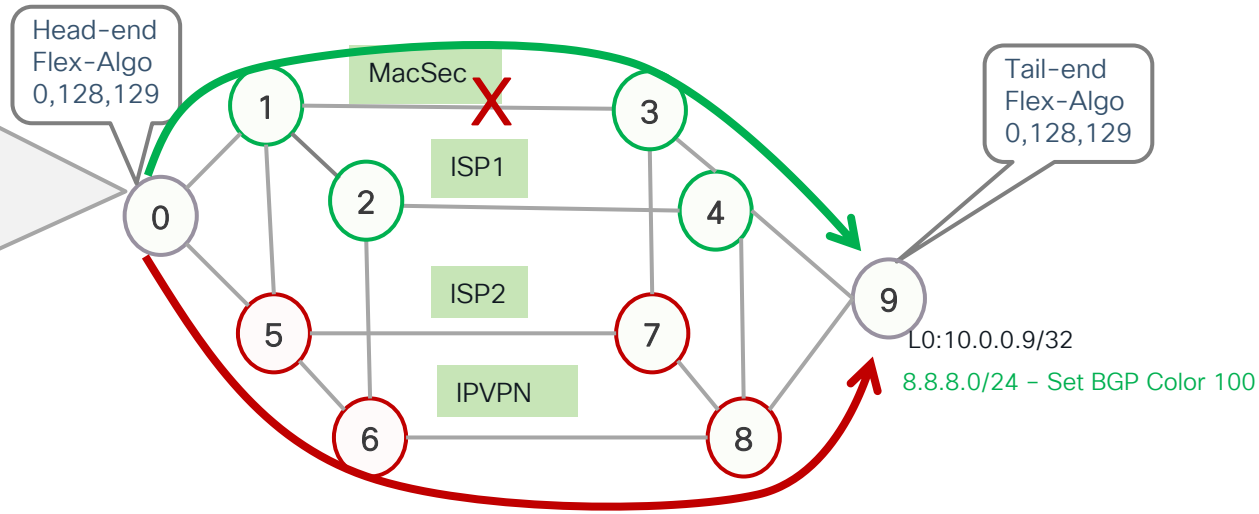
- Flex-algo 128, metric is IGP, excluded ISP1 link, Prefix color 100, label 16908
- Flex-algo 129, metric is delay, Prefix color 200, label 16909
- The BGP next-hop IP address and Prefix-SID IP address must be same Loopback

5.2 SR ODN Policy – Multi-Candidate Paths

- Automated Steering by leveraging IGP Flex-Algo

Node0 # IOS-XE

```
segment-routing traffic-eng
on-demand color 100
authorize
  candidate-paths
   preference 100
constraints
     segments
      dataplane mpls
      algorithm 128
!
preference 90
constraints
     segments
      dataplane mpls
      algorithm 129
```

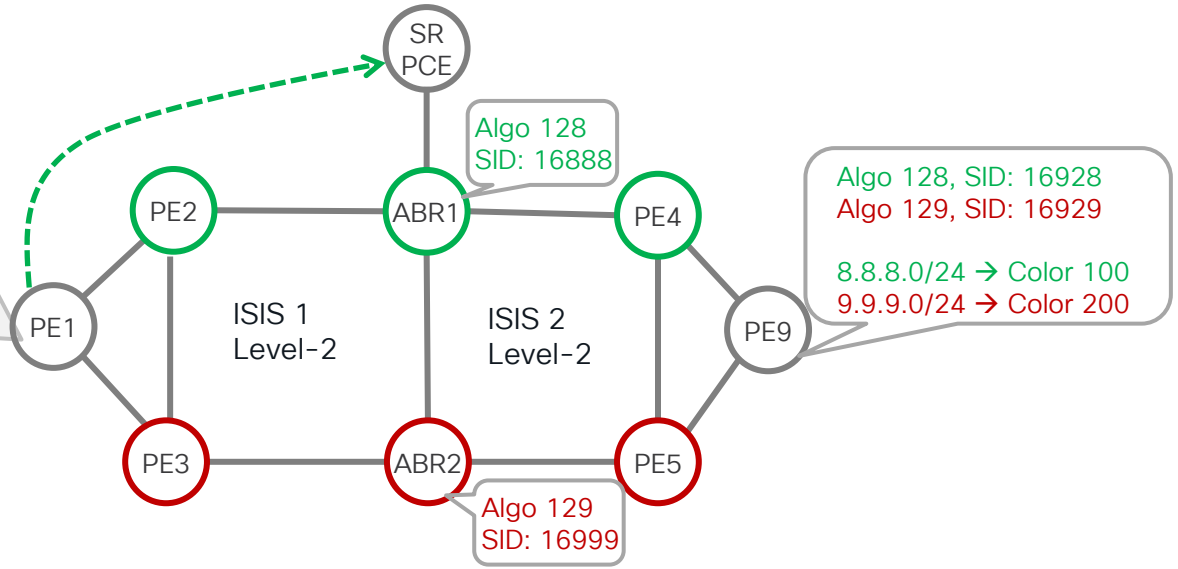


- Flex-algo 128, metric is IGP, excluded ISP1 link, Prefix color 100, label 16908
- Flex-algo 129, metric is IGP, exclude ISP1, Prefix color 200, label 16909

5.3 SRTE ODN Policy – Inter-Domain

IOS-XR:

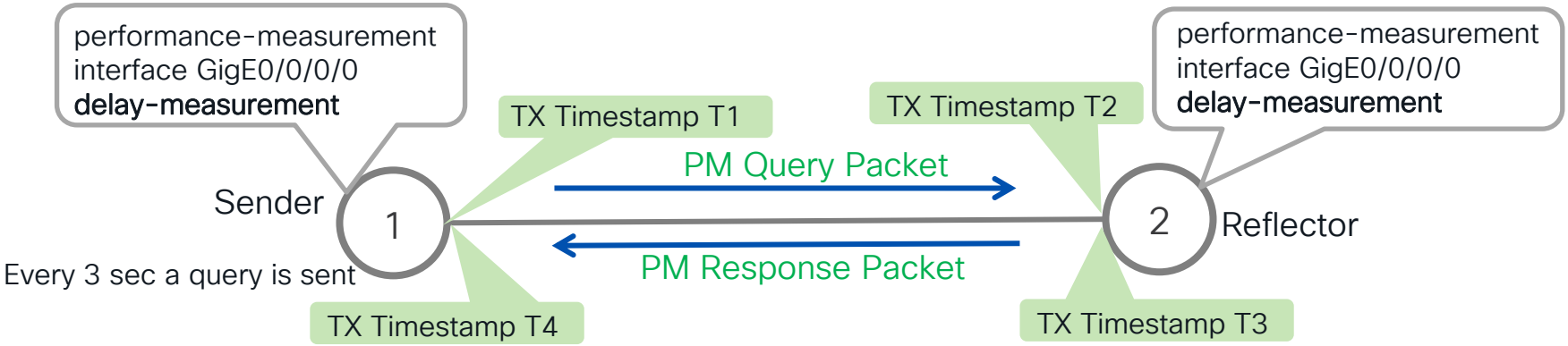
```
segment-routing
 traffic-eng
  on-demand color 100
   dynamic
    sid-algorithm 128
   !
on-demand color 200
   dynamic
bounds
     cumulative
      type igp <>
      type te <>
      type hopcount <>
      type latency <>
    sid-algorithm 129
   !
pcc
   source-address ipv4 192.168.0.1
   pce address ipv4 192.168.0.10
    precedence 100
```



- PCE supports Inter-domain policy using Flex-algo
- Inter-domain ODN Policy without PCE is in progress

Performance Measurement

Per-link Delay Measurement



- Router discover per-link DM and flood to IGP also can report to centralized controller via telemetry
- Two-way delay = (T2-T1)+(T4-T3) is by default, no clock synchronization is required
- One-way delay = Two-way delay/2, clock synchronization is required
- Sender and Reflector required HW Timestamping
- Two-Way Active Measurement Protocol (TWAMP-Light) uses RFC 5357 with IP/UDP encapsulation

Per-link delay Measurement

- Over a measurement internal
  - Minimum
  - Average
  - Maximum
  - Variance

Used as delay metric for SR ODN Policy

Not used by SR ODN Policy

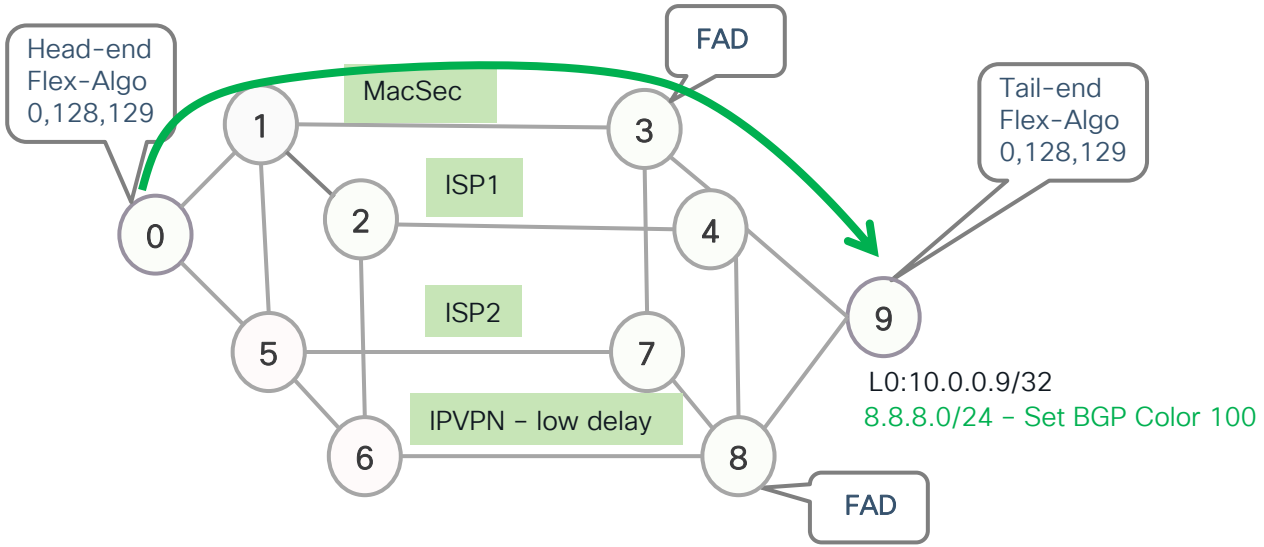
Use Cases

So why are we here?

- We want to be able to manipulate the IGP to our own specific needs using attribute or constraints that *we* decide are important?
  - Only use a route with a cumulative delay based on a measured per link delay
  - If we only want to use a secure path. Such as paths with MACsec only
  - Define a path traversing high speed links for bandwidth sensitive traffic
  - Only use a subset of the routers in your network

Use Case 1 – Secure Path

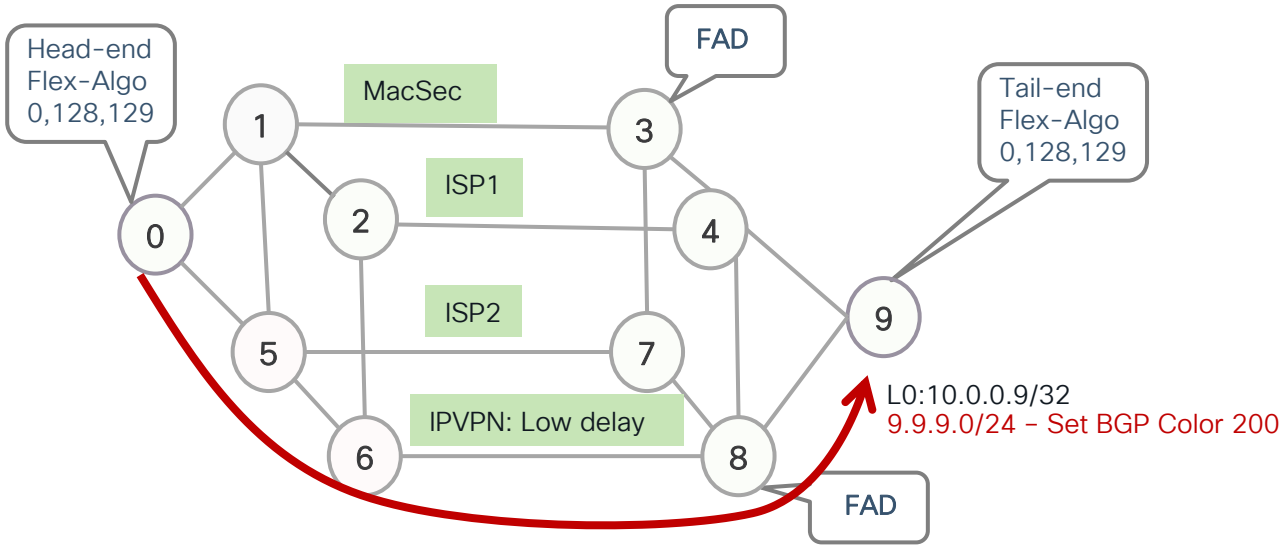
```
Node 0 IOS-XR#
router isis 1
 flex-algo 128
!
 address-family ipv4 unicast
  router-id Loopback0
  segment-routing mpls
 !
 interface Loopback0
  address-family ipv4 unicast
   prefix-sid absolute 16000
   prefix-sid algorithm 128 absolute 16800
!
segment-routing
 traffic-eng
!
 on-demand color 100
dynamic
    sid-algorithm 128
!
```



- All nodes support Algo 0, 128 and 129
- Algo 128 is associated with IGP metric and exclude ISP1, ISP2

Use Case 2- Real-time communications and applications

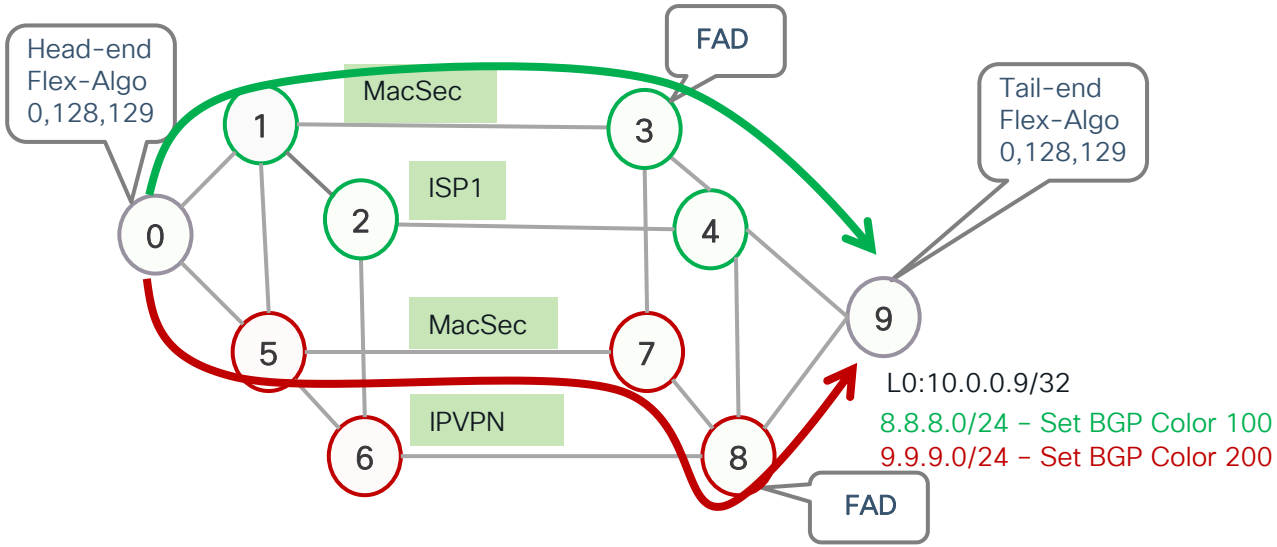
```
Node 0 IOS-XR#
router isis 1
 flex-algo 129
!
 address-family ipv4 unicast
  router-id Loopback0
  segment-routing mpls
 !
 interface Loopback0
  address-family ipv4 unicast
   prefix-sid absolute 16000
   prefix-sid algorithm 129 absolute 16900
!
performance-measurement
 interface GigabitEthernet0/0/0/X
  delay-measurement
  !
segment-routing
 traffic-eng
!
 on-demand color 200
dynamic
    sid-algorithm 129
!
```



- All nodes participating Algo 0, 128 and 129
- Algo 129 is associated with delay metric and no link affinity included or excluded
- Per-link delay measurement is flood to IGP

Use Case 3- Dual Plane / Multi-plane

```
Node 0 IOS-XR#
router isis 1
 flex-algo 128
!
 address-family ipv4 unicast
  router-id Loopback0
  segment-routing mpls
 !
 interface Loopback0
  address-family ipv4 unicast
   prefix-sid absolute 16000
   prefix-sid algorithm 128 absolute 16800
!
segment-routing
 traffic-eng
!
 on-demand color 100
dynamic
    sid-algorithm 128
!
on-demand color 200
dynamic
    sid-algorithm 129
```



- Algo 128 is associated with IGP metric and exclude ISP1
- Algo 129 is associated with IGP metric and exclude IPVPN

Additional Use Cases

- Only use a subset of the routers in your network
- Define a path traversing high speed links for bandwidth sensitive traffic

Flex-Algo support Highlights

- ISIS Flex-Algo
- OSPF Flex-Algo
- MPLS-PM: per-link delay measurement
- MPLS-PM: end-to-end SR Policy delay measurement
- SR Data Plane Monitoring (SR-DPM)
- Inter-domain ECMP and UCMP

SR ODN Support Highlights

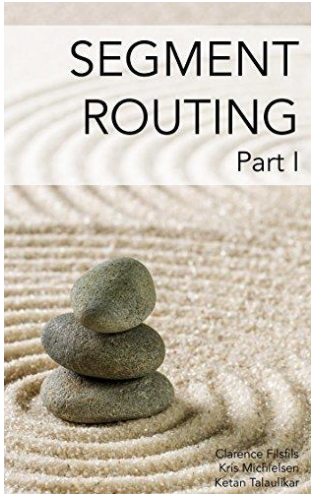
- The SR ODN Policy supports the following services:
  - IPv4 BGP global routes
  - IPv6 BGP global routes (6PE)
  - VPNv4
  - VPNv6 (6vPE)
  - EVPN-VPWS (single-homing)
  - EVPN-VPWS (multi-homing)
  - EVPN (single-homing/multi-homing)

IETF

- draft-ietf-spring-segment-routing
  - Prefix-SID per Algorithm
- draft-filsfils-spring-segment-routing-policy
  - SRTE architecture, ODN, AS
- draft-hegdeppsenak-isis-sr-flex-algo
  - Customization of Algo and consistency
- draft-ietf-isis-te-app
  - Used to flood Flex-Algo specific link affinities
- RFC7810 (IS-IS Traffic Engineering (TE) Metric Extensions)
  - Used to advertise extended TE metrics – e.g., link delay

Stay up-to-date

amazon.com/



segment-routing.net

linkedin.com/groups/8266623

twitter.com/SegmentRouting

facebook.com/SegmentRouting/

Technical Session Surveys

- Attendees who fill out a minimum of four session surveys and the overall event survey will get Cisco Live branded socks!
- Attendees will also earn 100 points in the Cisco Live Game for every survey completed.
- These points help you get on the leaderboard and increase your chances of winning daily and grand prizes.

Conclusion

Q&A

Thank you

Cisco Learning and Certifications

From technology training and team development to Cisco certifications and learning plans, let us help you empower your business and career. www.cisco.com/go/certs

Pay for Learning with Cisco Learning Credits

(CLCs) are prepaid training vouchers redeemed directly with Cisco.

Learn

Cisco U.
IT learning hub that guides teams and learners toward their goals

Cisco Digital Learning
Subscription-based product, technology, and certification training

Cisco Modeling Labs
Network simulation platform for design, testing, and troubleshooting

Cisco Learning Network
Resource community portal for certifications and learning

Train

Cisco Training Bootcamps
Intensive team & individual automation and technology training programs

Cisco Learning Partner Program
Authorized training partners supporting Cisco technology and career certifications

Cisco Instructor-led and Virtual Instructor-led training
Accelerated curriculum of product, technology, and certification courses

Certify

Cisco Certifications and Specialist Certifications
Award-winning certification program empowers students and IT Professionals to advance their technical careers

Cisco Guided Study Groups
180-day certification prep program with learning and support

Cisco Continuing Education Program
Recertification training options for Cisco certified individuals

Here at the event? Visit us at **The Learning and Certifications lounge at the World of Solutions**

Continue your education

- Visit the Cisco Showcase for related demos
- Book your one-on-one Meet the Engineer meeting
- Attend the interactive education with DevNet, Capture the Flag, and Walk-in Labs
- Visit the On-Demand Library for more sessions at www.CiscoLive.com/on-demand

cisco Live!

Additional Flex-Algo Slides

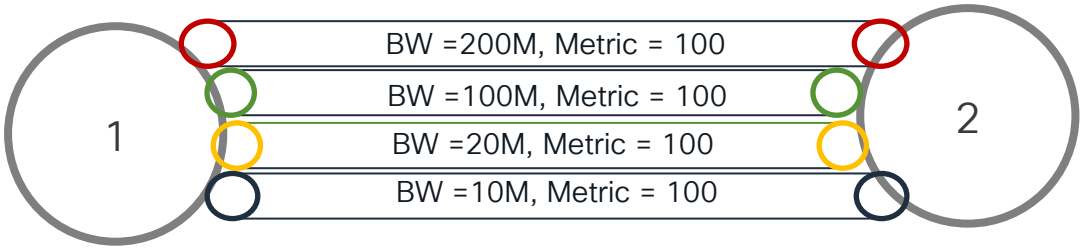
SR IGP Flex-Algo

- Leverages the SR-TE benefits of simplicity, automation and scalable
  - Automated sub-50msec FRR (TILFA), backup path honors Flex-Algo constraints
  - On-Demand Policy (ODN) and Automated Steering (AS)
  - Scale, no core state: state at the headend only
  - Supports Inter-domain latency and disjointed path
  - Use Prefix-SID label, no Adjacency label

- Example
  - Operator1 defines Flex-Algo(128) as “minimize IGP metric and avoid link-affinity *green*”
  - Operator2 defines Flex-Algo(128) as “minimize delay metric and avoid link-affinity *blue*”

UCMP – Unequal Cost Multi-Path

- UCMP = ECMP + Bandwidth, Per destination
  - Same IGP metric but different BW



Unequal Cost Multi-path

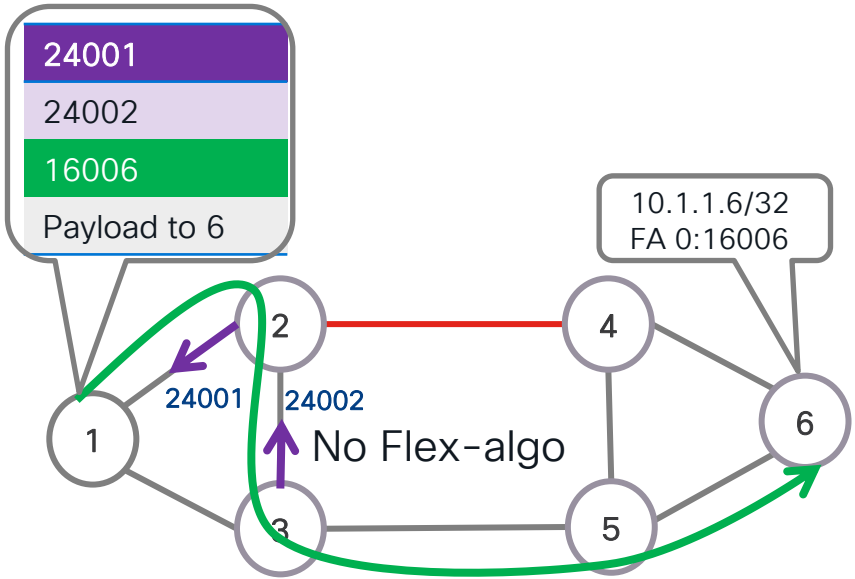
UCMP – Unequal Cost Multi-Path

- UCMP = ECMP + Bandwidth, Per destination
- Apply ”ucmp local” under ISIS process

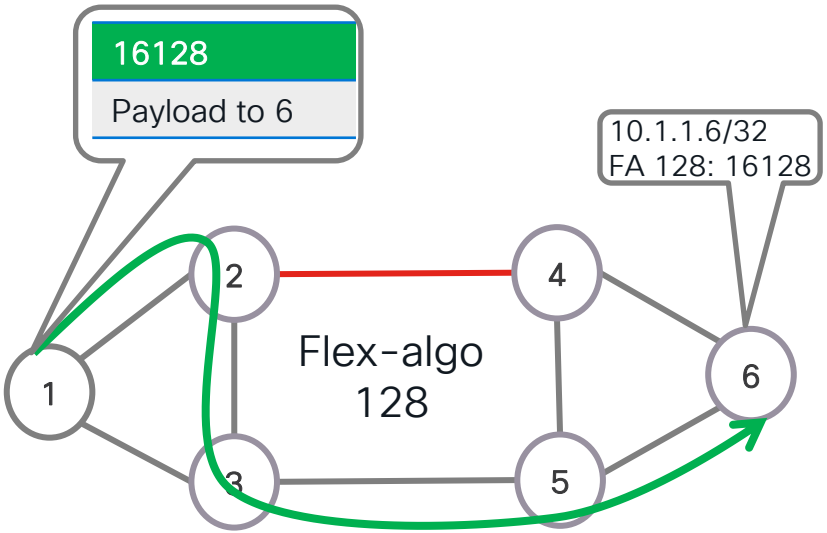
```
#show mpls forwarding-table labels 26042 detail | in Tu|load

Label      Label      or Tunnel Id     Switched      interface
                            0           Tu3002161   point2point
   Per-destination load-sharing, slots: 0
                            0           Tu3002141   point2point
   Per-destination load-sharing, slots: 1
                            0           Tu3002151   point2point
   Per-destination load-sharing, slots: 2
                            0           Tu3002131   point2point
   Per-destination load-sharing, slots: 3 6
                            0           Tu3002111   point2point
   Per-destination load-sharing, slots: 4 7 9 11 13 15
                            0           Tu3002121   point2point
   Per-destination load-sharing, slots: 5 8 10 12 14
```

Destination Prefix-SID

SRTE Path Computation without and with Flex-Algo



(a) Path Computation without Flex-algo



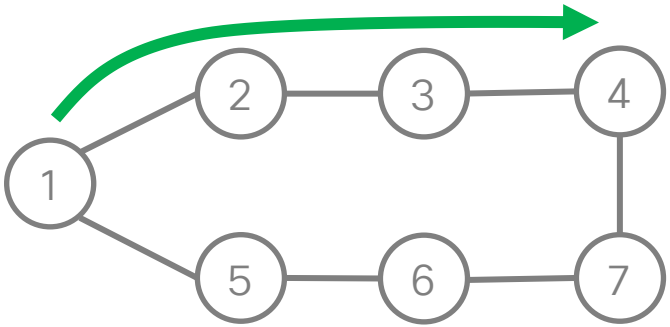
(b) Path Computation with Flex-algo, no adjacency label

Automated Steering – ODN Policy

- An on-demand SR policy is created dynamically for BGP or VPN
- On-Demand Next-hop (ODN) Policy
  - Intent based dynamic path computation
  - The ODN solution is solely related to dynamic instantiation of a candidate path
- An SR Policy is identified by three attributes, tuple:
  - Head-end: where the policy is instantiated
  - End-point: where the policy ends, the BGP next-hop address
  - Color: a numerical value assigned to a BGP prefix, represents an intent for SRTE policy

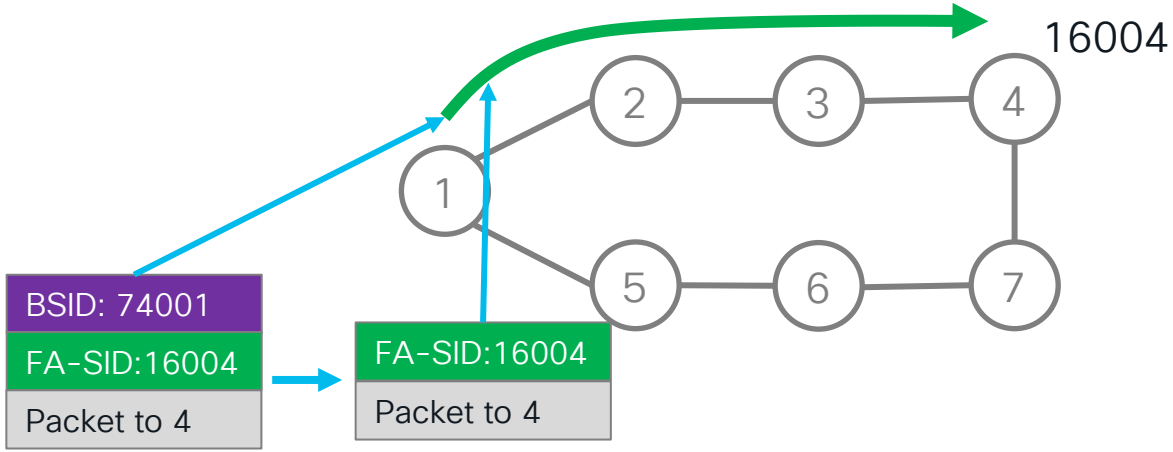
SR ODN Policy

Head-end: 1
Color: Green
End-point: 4



Binding SID (BSID)

- Binding SID is a local Segment ID or label bound to an SRTE Policy
- A BSID is associated with a single SRTE Policy
- By default, the head-end dynamically allocates the BSID, but the BSID can also be explicitly defined
- A BSID identifies a SRTE policy
  - Packet received with BSID as Top Label is steered into the SRTE Policy associated with the BSID
  - BSID label is popped, SRTE Flex-algo prefix-SID is pushed



BSID – Binding SID

- BSID is a local label that is auto generated, so right prefix can take the right SR ODN Policy

```
#sh segment-routing traffic-eng policy
end-point ipv4 10.101.1.4

SR-TE policy database
---------------------
Color: 850, End-point: 10.101.1.4
 Binding SID: 76623
```

```
#  sh bgp vpnv4 unicast vrf CUSTOMER1 11.103.1.1

10.101.1.4 C:850 (bsid:76623) (metric 200040) from
10.100.0.1 (10.101.1.4)

SR policy color 850, up, registered, bsid 76623, if-handle
0x3c0080d4 Local, (received-only)

10.101.1.4 C:850 (bsid:76623) (metric 200040) from
10.100.0.1 (10.101.1.4
```

SR ODN Policy Color

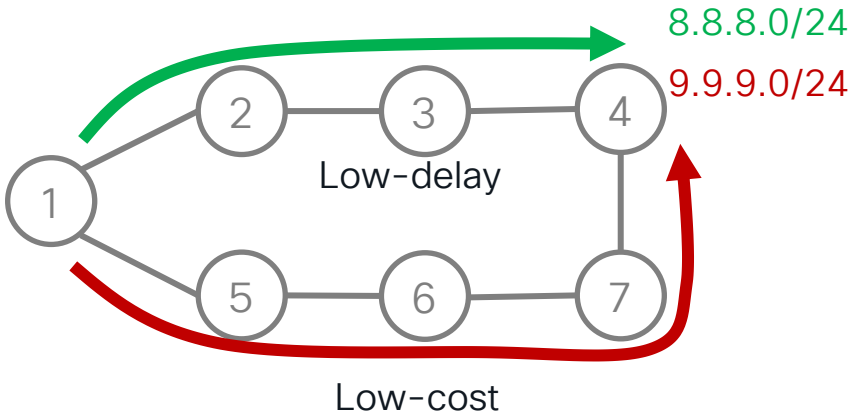
- Each SR Policy has a color
  - BGP color is used to provide certain treatment (SLA) to some applications by SR ODN policy
  - Each SRTE ODN Policy has a unique triplet (H,C,E)
  - A prefix with multiple colors will choose highest numerical value to steer traffic

SR ODN Policy

1,green,4
steer 8.8.8.0/24 via low delay path

SR ODN Policy

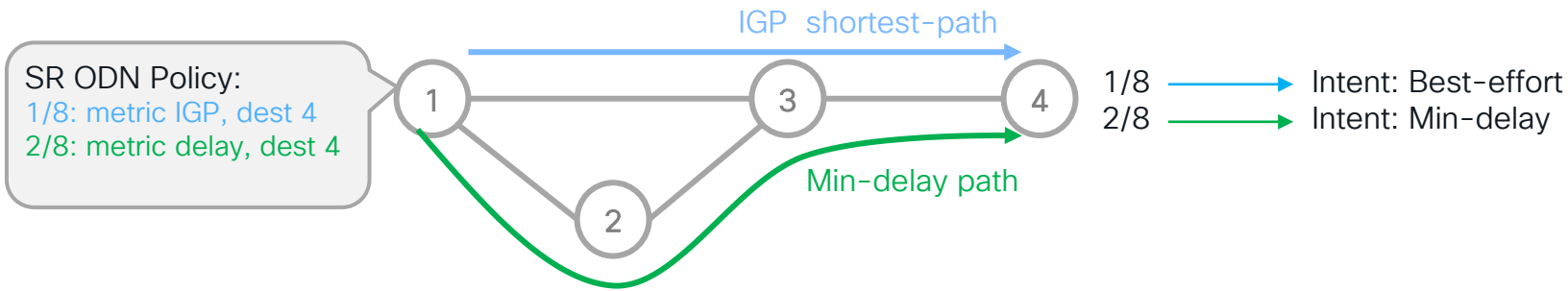
1,red,4
steer 9.9.9.0/24 via Low-cost path



The BGP color extended community is specified in RFC5512

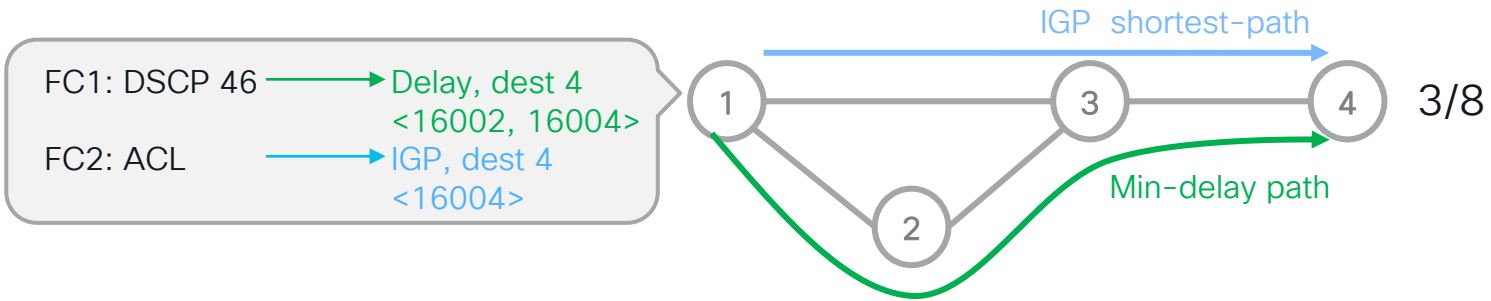
Automated Steering - Per-Destination Policy (PDP)

- Per-destination policy
  - Steer traffic based on next-hop and color of a BGP service route
  - Color is a BGP extended community attribute
  - Color is used for transport SLA indicator, for instance min-delay or min-cost



Automated Steering - Per-Flow Policy (PFP)

- Per-flow policy
  - Steer traffic based on incoming packets classification (IPP, DSCP, ACL, EXP etc.)
  - Then set local Forward-Class up to 8, range 0-7
  - An ingress PBR policy applied to an input interface



Automated Steering - Per-Flow Policy (PFP)

Traffic Classification:

```
class-map type traffic match-any MinDelay
match dscp46
end-class-map
!
class-map type traffic match-any PremiumHosts
match access-group ipv4 PrioHosts
end-class-map
```

```
policy-map type pbr  MyPFP
class type traffic MinDelay
set forward-class 1
!
class type traffic PremiumHosts
set forward-class 2
!
class type traffic class-default
set forward-class 0
```

```
interface GigabitEthernet0/0/0/0
description PE_Ingress_Interface
service-policy type pbr input MyPFP
```

Per-Flow Policy:

```
segment-routing
 traffic-eng

  on-demand color 10
   dynamic
    sid-algorithm 128
   !
```

```
on-demand color 20
   dynamic
    sid-algorithm 129
!
on-demand color 30
   dynamic
    sid-algorithm 130
```

```
on-demand color 1000
  per-flow
    forward-class 0 color 10
    forward-class 1 color 20
    forward-class 2 color 30
```

Headend SRTE DB – IGP Config

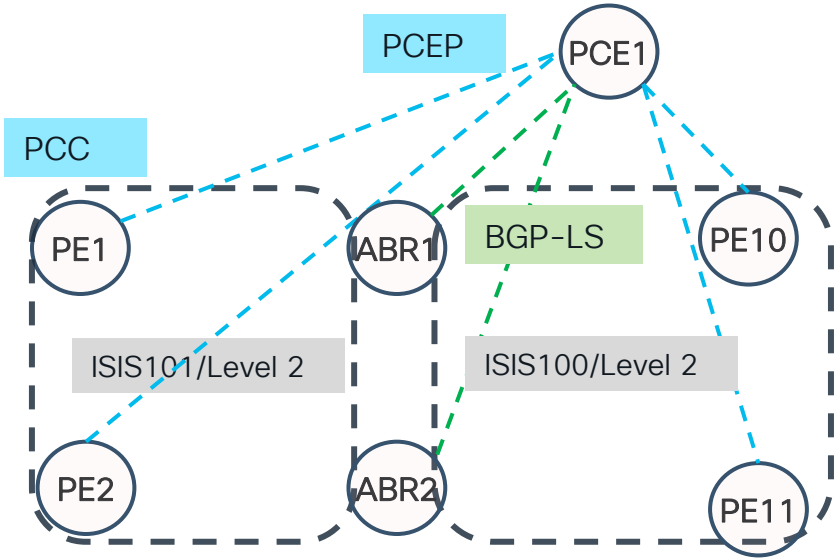
- Enable the following command under ISIS/OSPF to feed the SRTE DB on the head-end:

```
router isis 1
distribute link-state
```

```
router ospf 1
distribute link-state
```

PCE – Path Computation Element

- IGP principle is to keep the node and link-state info within its own IGP area
- ABR redistribute node and link-state info from ISIS to BGP-LS address-family
- PE run path computation element protocol (PCEP) with the PCE for inter-domain node and link-state info



PCC: Path computation client
PCE: Path computation element
PCEP: PCE protocol
BGP-LS: BGP link-state

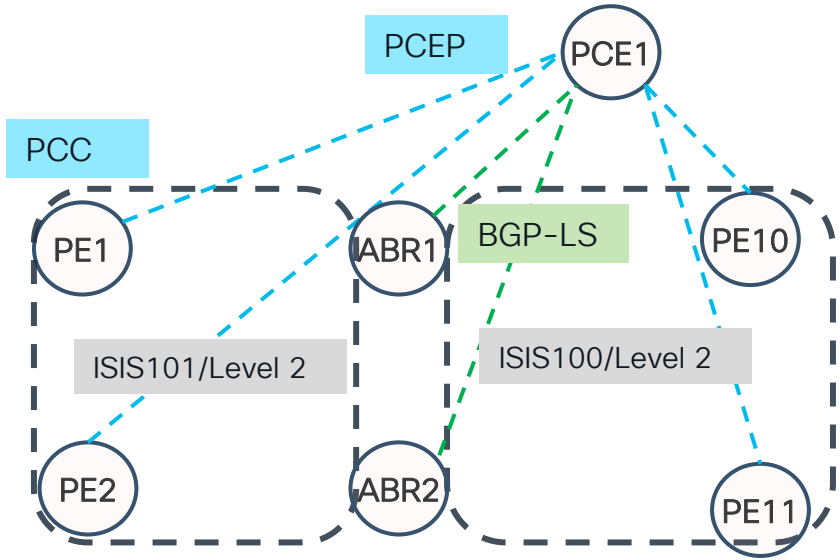
PCE – Path Computation Element

IOS-XR:

PCC#
```
segment-routing
 traffic-eng
 pcc
   source-address ipv4 x.x.x.x
   pce address ipv4 x.x.x.y
    precedence 100
```

ABR#
```
router isis 1
 distribute link-state instance-id 1 level X
```

PCE#
```
pce
 address ipv4 x.x.x.z
```



PCC: Path computation client
PCE: Path computation element
PCEP: PCE protocol
BGP-LS: BGP link-state

ODN Policy – Per-Destination (Inter-Domain)

IOS-XR:

```
PE1#show segment-routing traffic-eng policy color 100
SR-TE policy database
---------------------
Color: 100, End-point: 192.168.0.9
  Name: srte_c_100_ep_192.168.0.9
  Status:
    Admin: up  Operational: up
Candidate-paths:
    Preference: 200 (BGP ODN) (inactive)
    Preference: 100 (BGP ODN) (active)
      Requested BSID: dynamic
Constraints:
       Prefix-SID Algorithm: 128
 Dynamic (pce 192.168.0.10) (valid)
         16888 [Prefix-SID, 192.168.0.11]
         16928 [Prefix-SID, 192.168.0.9]
  Attributes:
    Binding SID: 24010

PE1#  sh bgp vrf GREEN 8.8.8.1
BGP routing table entry for 8.8.8.0/24, Route Distinguisher: 1:1
Local
    192.168.0.9 C:100 (bsid:24010) (metric 120) from 192.168.0.10
(192.168.0.9)
```

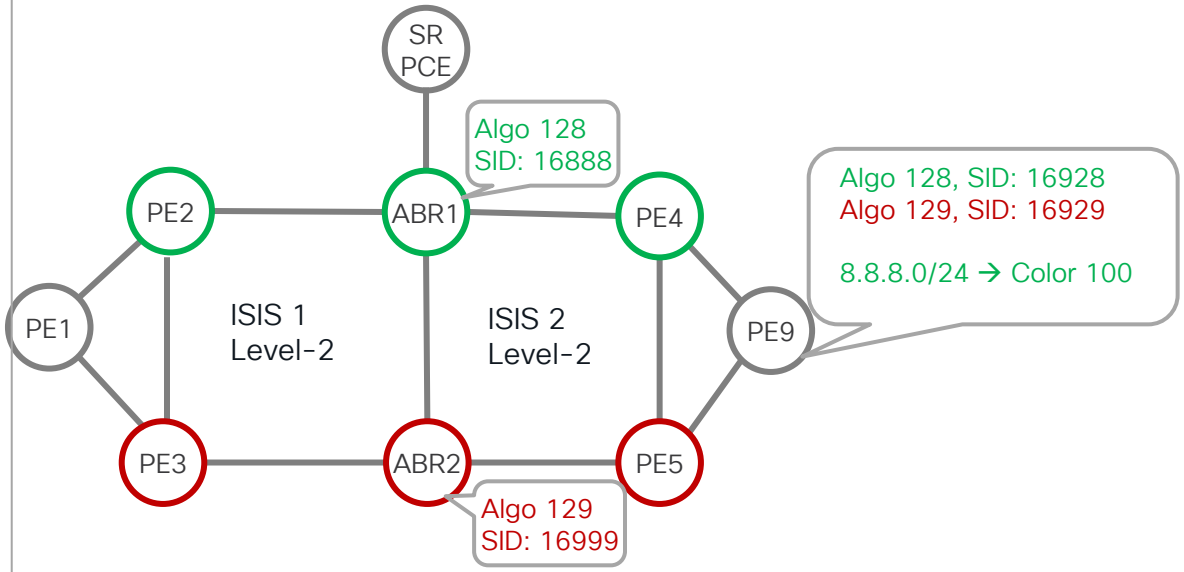
Pref 200 is failed because for Inter-domain ODN policy, the local DB has no link info for other domain and then it moved to PCE with pref 100 and succeed.

For inter-domain it uses two prefix-SIDs, 1st SID for head-end to ABR router and 2nd SID for ABR to tail-end.

SRTE ODN Policy– Multi-Candidate Paths

IOS-XE:

```
segment-routing traffic-eng
 !
 on-demand color 100
  authorize
  candidate-paths
   preference 200
    constraints
     segments
      dataplane mpls
      algorithm 128
     !
dynamic
      pcep
    !
   preference 100
    constraints
     segments
      dataplane mpls
      algorithm 129
     !
dynamic
      pcep
    !
pcc
  pce address 192.168.0.10 source-address 192.168.0.1
```



- For network 8.8.8.0/24, Flex-algo 128 Green nodes are primary
- path and Flex-algo 129 Red nodes are backup path.

SRTE ODN Policy– Multi-Candidate Paths

IOS-XE:

```
PE1#sh segment-routing traffic-eng policy all

Name: *192.168.0.9|100 (Color: 100 End-point: 192.168.0.9))
  Candidate-paths:
    Preference 200 (BGP):
      Constraints:
        Algorithm: 128
      Dynamic (pce 192.168.0.10) (active)
          16011 [Prefix-SID, 192.168.0.11]
          16008 [Prefix-SID, 192.168.0.9]
Preference 100 (BGP):
      Constraints:
        Algorithm: 129
      Dynamic (pce 192.168.0.10) (inactive)
          16012 [Prefix-SID, 192.168.0.12]
          16009 [Prefix-SID, 192.168.0.9]
  Attributes:
    Binding SID: 22
      Allocation mode: dynamic
      State: Programmed
  IPv6 caps enabled
```

IOS-XE:

```
PE1#sh bgp vrf GREEN 8.8.8.1
BGP routing table entry for 1:1:8.8.8.0/24, version 64
Paths: (1 available, best #1, table GREEN)
  Not advertised to any peer
  Refresh Epoch 1
  Local
    192.168.0.9 (metric 120) (via default) from 192.168.0.10 (192.168.0.10)
      Origin incomplete, metric 0, localpref 100, valid, internal, best
      Extended Community: RT:1:10 Color:100
      Originator: 192.168.0.9, Cluster list: 192.168.0.10
      mpls labels in/out nolabel/24000
      binding SID: 22 (color - 100) (state - UP)
      rx pathid: 0, tx pathid: 0x0
      Updated on Apr 21 2022 01:49:24 UTC
```

Default

- Every 3 second, a query
  - a two-way query is sent
- Every 30 seconds, a probe
  - min, avg, max, var are computed over the last 10 queries
  - Last-Probe EDT trigger with (min, avg, max, var)
- Every 120 seconds, an aggregation
  - min, avg, max, var over the last 4 probes are computed
  - Last-Aggregation Even Driven Telemetry trigger with (min, avg, max, var)
  - IF [abs(min-F.min)/F.min >= 10%] and [abs(min-F.min)>=1000usec]
    THEN an LSDB change is triggered to flood the new link delay values
    a last-advertisement EDT is triggered with these values

F.min is the last flooded value of min-delay. This is what the rest of the network thinks of this link min delay.

Minimum delay is of interest for SRTE

- Minimum delay provides the propagation delay
  - fiber length / speed of light
- A property of the topology
  - with awareness of DWDM circuit change
- SRTE (Policy or Flex-Algo) can optimize on min delay

Average, Max and Variance are dealt with by QoS

- Depends on congestion
  - (traffic burst over line rate) / line rate
- Highly variable at any time scale
- Not controlled by routing optimization
- Controller by QoS
  - Priority queue, WRR, WFQ…
  - Tail-Drop, RED…


cisco Live!
ALL IN