Understanding BGP Dynamic Segment Routing Traffic Engineering (SR-TE)

Contents

Introduction
Prerequisites
Requirements
Components Used
Background Information
Topology diagram
Initial Configurations
Configuring BGP Dynamic SR-TE
Verifications
Summary

Introduction

This document describes the aspects of understanding, configuring, and verifying the BGP Dynamic SR-TE feature in IOS-XR.

Contributed by Elvin Arias, Cisco TAC Engineer.

Prerequisites

There are no prerequisites for this document.

Requirements

There are no specific requirements for this document.

Components Used

The information in this document is based on Cisco IOS-XR® and IOS-XE®.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Background Information

Segment Routing Traffic Engineering (SR-TE) provides the capabilities to steer traffic through an SR-enabled core without state creation and maintenance (stateless). An SR-TE policy is expressed as a list of segments that specifies a path, called Segment ID (SID) list. No signaling is required as state is in the packet and SID list are processed as a set of instructions by the transit routers.
With Dynamic BGP SR-TE we can generate automatic SR-TE policies based on arbitrary criteria like communities signaled by a router participating in a Segment Routing network. To be able to meet service level assurance (SLAs) of customer’s applications and compute paths based on specific requirements, we can generate automatic SR-TE policies for a given IP subnet or services by setting communities and triggering these policies accordingly.

**Spoiler**

**Note:** Matching criteria other than communities is also supported to create dynamic SR-TE policies.

Note: Matching criteria other than communities is also supported to create dynamic SR-TE policies.

A common application for this feature is in MPLS L3VPN environments, where network administrator can trigger automatic SR-TE tunnel policies to route traffic based on specific constraints (delay, bandwidth, etc.). For below demonstrations, we will create a L3VPN service connecting XR1 and XR5 and trigger auto-tunnels on XR2 (headend) based on a particular community set on XR4 (tail end) on MP-BGP.

**Topology diagram**

![Topology diagram]

**Initial Configurations**

We have enabled L3VPN, Segment Routing, and SR-TE basic configurations.

 XR1
hostname XR1
logging console debugging
interface Loopback0
    ipv4 address 1.1.1.1 255.255.255.255
 !
interface GigabitEthernet0/0/0/0.12
    ipv4 address 12.0.0.1 255.255.255.0
    encapsulation dot1q 12
 !
route-policy PASS
    pass
end-policy
 !
router bgp 200
interface GigabitEthernet0/0/0/0.45 ipv4 address 45.0.0.5 255.255.255.0 encapsulation dot1q 45 ! route-policy PASS pass end-policy ! router bgp 200 bgp router-id 5.5.5.5 bgp unsafe-ebgp-policy address-family ipv4 unicast network 5.5.5.5/32 network 5.5.5.55/32 ! neighbor 45.0.0.4 remote-as 100 address-family ipv4 unicast route-policy PASS in route-policy PASS out ! ! mpls oam ! end

XR6
hostname XR6 logging console debugging interface Loopback0 ipv4 address 6.6.6.6 255.255.255.255 ! interface GigabitEthernet0/0/0/0.26 ipv4 address 26.0.0.6 255.255.255.0 encapsulation dot1q 26 ! interface GigabitEthernet0/0/0/0.46 ipv4 address 46.0.0.6 255.255.255.0 encapsulation dot1q 46 ! router ospf 1 segment-routing mpls segment-routing forwarding mpls segment-routing sr-prefer address-family ipv4 area 0 mpls traffic-eng interface Loopback0 prefix-sid index 6 ! interface GigabitEthernet0/0/0/0.26 cost 200 network point-to-point ! interface GigabitEthernet0/0/0/0.46 cost 200 network point-to-point ! mpls traffic-eng router-id Loopback0 ! mpls oam ! mpls traffic-eng interface GigabitEthernet0/0/0/0.26 admin-weight 1 ! interface GigabitEthernet0/0/0/0.46 admin-weight 1 ! ! end

XR2 and XR4 (PEs) have built an LSP using Segment Routing, this can be verified by using MPLS ping for the corresponding Segment Routing FEC. For this scenario, we have two possible paths for transporting the L3VPN traffic from XR1 to XR5:

Regular LSP path: XR1 -> XR2 -> XR3 -> XR4 -> XR5

Low latency LSP path: XR1 -> XR2 -> XR6 -> XR4 -> XR5

Initially, all traffic between XR1 and XR5 is being routed through XR3 via the regular LSP path due to lower IGP cost, we can confirm both LSPs and connectivity as per below verifications. IGP cost to reach XR4 from XR2 via XR3 is 201 versus 401 via XR6. Even though the path via XR3 has a better path metric, low latency services on VRF BLUE must be routed through the path via XR6.

RP/0/0/CPU0:XR2#ping mpls ipv4 4.4.4.4/32 fec-type generic verbose

Sending 5, 100-byte MPLS Echos to 4.4.4.4/32, timeout is 2 seconds, send interval is 0 msec:


Type escape sequence to abort.

! size 100, reply addr 34.0.0.4, return code 3
! size 100, reply addr 34.0.0.4, return code 3
! size 100, reply addr 34.0.0.4, return code 3
! size 100, reply addr 34.0.0.4, return code 3
! size 100, reply addr 34.0.0.4, return code 3

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/10 ms

Spoiler
Note: When using ping MPLS application in Segment Routing we must use Nil-FEC or generic FEC.

Note: When using ping MPLS application in Segment Routing we must use Nil-FEC or generic FEC.

If we verify the L3VPN services on XR1, we can confirm that we have reachability to XR5
loopback 5.5.5.5/32 and 5.5.5.55/32 respectively via the regular LSP path. Basic L3VPN services are enabled in the SR MPLS core.

RP/0/0/CPU0:XR1#ping 5.5.5.5 source 1.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 5.5.5.5, timeout is 2 seconds:
!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/7/9 ms

RP/0/0/CPU0:XR1#ping 5.5.5.55 source 1.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 5.5.5.55, timeout is 2 seconds:
!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/7/9 ms

RP/0/0/CPU0:XR1#traceroute 5.5.5.5 source 1.1.1.1

Type escape sequence to abort.
Tracing the route to 5.5.5.5
1  12.0.0.2 9 msec 0 msec 0 msec
2  23.0.0.3 [MPLS: Labels 16004/24002 Exp 0] 0 msec 0 msec 0 msec
3  34.0.0.4 [MPLS: Label 24002 Exp 0] 0 msec 0 msec 0 msec
4  45.0.0.5 0 msec * 0 msec

RP/0/0/CPU0:XR1#traceroute 5.5.5.55 source 1.1.1.1

Type escape sequence to abort.
Tracing the route to 5.5.5.55
1  12.0.0.2 9 msec 0 msec 0 msec
2  23.0.0.3 [MPLS: Labels 16004/24005 Exp 0] 0 msec 0 msec 0 msec
3  34.0.0.4 [MPLS: Label 24005 Exp 0] 0 msec 0 msec 0 msec
4  45.0.0.5 0 msec * 0 msec

As observed, all traffic on VRF BLUE is going through the regular LSP path XR1 -> XR2 -> XR3 -> XR4 -> XR5.

Configuring BGP Dynamic SR-TE

For this example we will configure XR4 (tail end) to insert community 1:1 and send it to XR2 to signal the creation of an SR-TE policy for prefix 5.5.5.55/32 on VRF BLUE. SR-TE policy path selection will be set to take the low latency path instead of the regular LSP, we will do this by selecting the lowest TE metric (Admin Weight) via XR6. Total TE metric (admin weight) via XR6 is 2, as admin weights have been set to 1 on outgoing interfaces towards XR4 (tail end) via XR6 as seen in the reference topology diagram and initial configurations.

In order to create the dynamic SR-TE policies, we need to configure what loopback will be used as source and what is the dynamic tunnel range that the headend will use generate the tunnels, this configuration is required at the headend of the SR-TE policy XR2. We will set the tunnel range to a minimum of 500 and a maximum of 500, effectively creating a single SR-TE tunnel and the source loopback to loopback 0 at the headend for the tunnel.
On XR4, we will set the community 1:1 and apply it on the VRF BLUE prefix 5.5.55/32, this will allow it to insert the community in the BGP update.

XR4
route-policy COMMUNITY_1:1
  # 1:1 Community
  if destination in (5.5.55/32) then
    set community (1:1)
  endif
  pass
end-policy
!
routing bgp 100
  vrf BLUE
  !
  neighbor 45.0.0.5
  address-family ipv4 unicast
    route-policy COMMUNITY_1:1 in
  !
end

Verifying XR2 (headend) we can see it has the community 1:1 set on the VPNv4 updates received from XR4.

RP/0/0/CP00:XR2# show bgp vrf BLUE 5.5.55.32 detail
BGP routing table entry for 5.5.55.32, Route Distinguisher: 1:1 Versions: Process bRIB/RIB
SendTblVer Speaker 36 36 Flags: 0x00043001+0x00000200; Last Modified: Nov 23 17:50:59.798 for
00:02:53 Paths: (1 available, best #1) Advertised to CE peers (in unique update groups):
12.0.0.1 Path #1: Received by speaker 0 Flags: 0x4000000085060005, import: 0x9f Advertised to CE
peers (in unique update groups): 12.0.0.1 200 4.4.4.4 (metric 201) from 4.4.4.4 (4.4.4.4)
Received Label 24005 Origin IGP, metric 0, localpref 100, valid, internal, best, group-best,
import-candidate, imported Received Path ID 0, Local Path ID 0, version 36 Community: 1:1
  Extended community: RT:1:1
  Source API: VPNv4 Unicast, Source VRF: BLUE, Source Route Distinguisher: 1:1

On XR2 (headend) we will create an RPL route policy matching the community 1:1 and setting the corresponding attribute-set for MPLS traffic-engineering. After the policy is set, we can go to the MPLS-TE configuration stanza and set the corresponding attribute-set for the SR-TE policy and indicate what is the path selection criteria, which are Segment Routing and TE metric in this case since we want to choose the path via the lowest administrative weight via XR6.

XR2
route-policy DYN_BGP_SR-TE
  # Matches community 1:1
  if community matches-every (1:1) then
    set mpls traffic-eng attributeset DYN_SR-TE_POLICIES
  endif
  pass
end-policy
Verifications

Once completed, we can observe that tunnel-te 500 interface has been dynamically created for the specified range.

BGP RIB indicates that the "DYN_SR-TE_POLICIES" policy is attached to the prefix, which means traffic must be routed according to the policy.

If we verify the BGP RIB for the prefix 5.5.5.55/32 in detail we can see the control plane information that will be referenced to generate the SR-TE tunnel.
We can see that the tunnel policy is in **up** state and **registered**. The binding SID assigned is 24000, this binding SID can be used to verify what tunnel is used for this particular prefix. As observed before, tunnel-te500 was created and installed in the LFIB.

### SR-TE policy on XR2 (headend)

SR-TE policy on XR2 (headend) will have the following properties from a control plane and data plane perspective to forward traffic. Also state information of the SR-TE tunnel can be seen as per below output, which must match with previous verifications.
show mpls traffic-eng tunnels segment-routing p2p 500

Name: tunnel-te500  Destination: 4.4.4.4  Ifhandle:0x130 (auto-tunnel for BGP default)
Signalled-Name: auto_XR2_t500
Status:
  Admin:    up  Oper:   up  Path:  valid  Signalling: connected
  path option 10, (Segment-Routing) type dynamic  (Basis for Setup, path weight 2)
  G-PID: 0x0800 (derived from egress interface properties)
  Bandwidth Requested: 0 kbps  CT0
  Creation Time: Fri Nov 23 17:55:23 2018  (00:09:01 ago)
Config Parameters:
  Bandwidth:        0 kbps  (CT0) Priority:  7  7 Affinity: 0x0/0x0
  Metric Type: TE (interface)
  Path Selection:
    Tiebreaker: Min-fill (default)
    Protection: Unprotected Adjacency
    Hop-limit: disabled
    Cost-limit: disabled
    Path-invalidation timeout: 10000 msec (default), Action: Tear (default)
  AutoRoute: disabled  LockDown: disabled  Policy class: not set
  Forward class: 0 (default)
  Forwarding-Adjacency: disabled
  Autoroute Destinations: 0
  Loadshare:          0 equal loadshares
  Auto-bw: disabled
  Path Protection: Not Enabled
  Attribute-set: DYN_SR-TE_POLICIES (type p2p-te)
  BFD Fast Detection: Disabled
  Reoptimization after affinity failure: Enabled
  SRLG discovery: Disabled
History:
  Tunnel has been up for: 00:09:01 (since Fri Nov 23 17:55:23 UTC 2018)
  Current LSP:
    Uptime: 00:09:01 (since Fri Nov 23 17:55:23 UTC 2018)
  Reopt. LSP:
    Last Failure:
      LSP not signalled, identical to the [CURRENT] LSP
      Date/Time: Fri Nov 23 17:56:53 UTC 2018  [00:07:31 ago]
Segment-Routing Path Info (OSPF 1 area 0)
  Segment0[Link]: 26.0.0.2 - 26.0.0.6, Label: 24005
  Segment1[Link]: 46.0.0.6 - 46.0.0.4, Label: 24003
Displayed 1 (of 1) heads, 0 (of 0) midpoints, 0 (of 0) tails
Displayed 1 up, 0 down, 0 recovering, 0 recovered heads

Checking the prefix directly on VRF BLUE RIB, we can confirm that binding SID 24000 was assigned to the prefix.

show route vrf BLUE 5.5.5.55/32 detail

Routing entry for 5.5.5.55/32
  Known via "bgp 100", distance 200, metric 0
  Tag 200, type internal
  Installed Nov 23 17:55:23.267 for 00:10:38
Routing Descriptor Blocks
  4.4.4.4, from 4.4.4.4
     Nexthop in Vrf: "default", Table: "default", IPv4 Unicast, Table Id: 0xe0000000
     Route metric is 0
     Label: 0x5dc5 (24005)
FIB for VRF BLUE indicates that forwarding for this prefix is being done via tunnel-te 500 according to our BGP dynamic SR-TE policy.

On XR1 we can verify connectivity and confirm that traffic is going through tunnel-te 500 via low latency path via XR6.

XR2 counters increase for the tunnel-te500 which corresponds to our SR-TE policy.
Path for prefix 5.5.5.5/32 is still going through the regular LSP path via XR3 as seen below.

```
RP/0/0/CPU0:XR1#traceroute 5.5.5.5 source 1.1.1.1
Type escape sequence to abort.
Tracing the route to 5.5.5.5

1  12.0.0.2 0 msec 0 msec 0 msec
2  23.0.0.3 [MPLS: Labels 16004/24002 Exp 0] 0 msec 0 msec 0 msec
3  34.0.0.4 [MPLS: Label 24002 Exp 0] 0 msec 0 msec 0 msec
4  45.0.0.5 0 msec * 0 msec
```

Summary

BGP Dynamic SR-TE offers granularity and automatic enforcing of routing policies for the purpose of traffic engineering in the SR enabled core. Automatic tunnel creation can be triggered based on arbitrary criteria, which can allow network administrators to easily create traffic patterns that meets customer's application requirements.