



June 9-13, 2019 • San Diego, CA

Programmability with SD-WAN

DEVWKS-1671

Jairo Leon
Marcelo Garcia

Table of Contents

Table of Contents.....	2
Learning Objectives	3
Network Diagram.....	4
IP Address Plan.....	4
Management IP Address Table	4
Hosts available in each site (per VPN) Table	5
DC1-Edge1 IP Address.....	5
DC1-Edge2 IP Address.....	5
Branch 3 IP Address	5
MPLS PE ROUTER (Emulating MPLS Network) Table.....	5
INET PE ROUTER (Emulating Internet Network) Table.....	6
INET ROUTER (Emulating LTE Access – Internet Network) Table.....	6
INTERNET ACCESS GATEWAY Table.....	6
Use Case 1 / LAB1 – Change system configuration with vMANAGE APIs and feature templates	7
Postman collection link:.....	7
Quick postman review:	7
Step 1: Authentication.....	8
Step 2: Get the templateID associated to a deviceID.....	8
Step 3: Create BANNER template.....	10
Step 4: Associate the new banner template with the device template	12
Step 5: Re-apply the template associated with BR3-vEDGE	15
Step 6: Verification	18
Use Case 2 / LAB 2 - Policies	20
Step 1: Login into vManage and validate the Policies.....	20
Step 2: Create application aware policy.....	22
Step 3: Create a template policy using the application aware policy (configured in step 1) and a prefix list.....	24
Step 4: Activate policy.....	27
Step 5: Verification	28
Step 6: Deactivate policy using Postman (OPTIONAL)	29
References.....	30

Learning Objectives

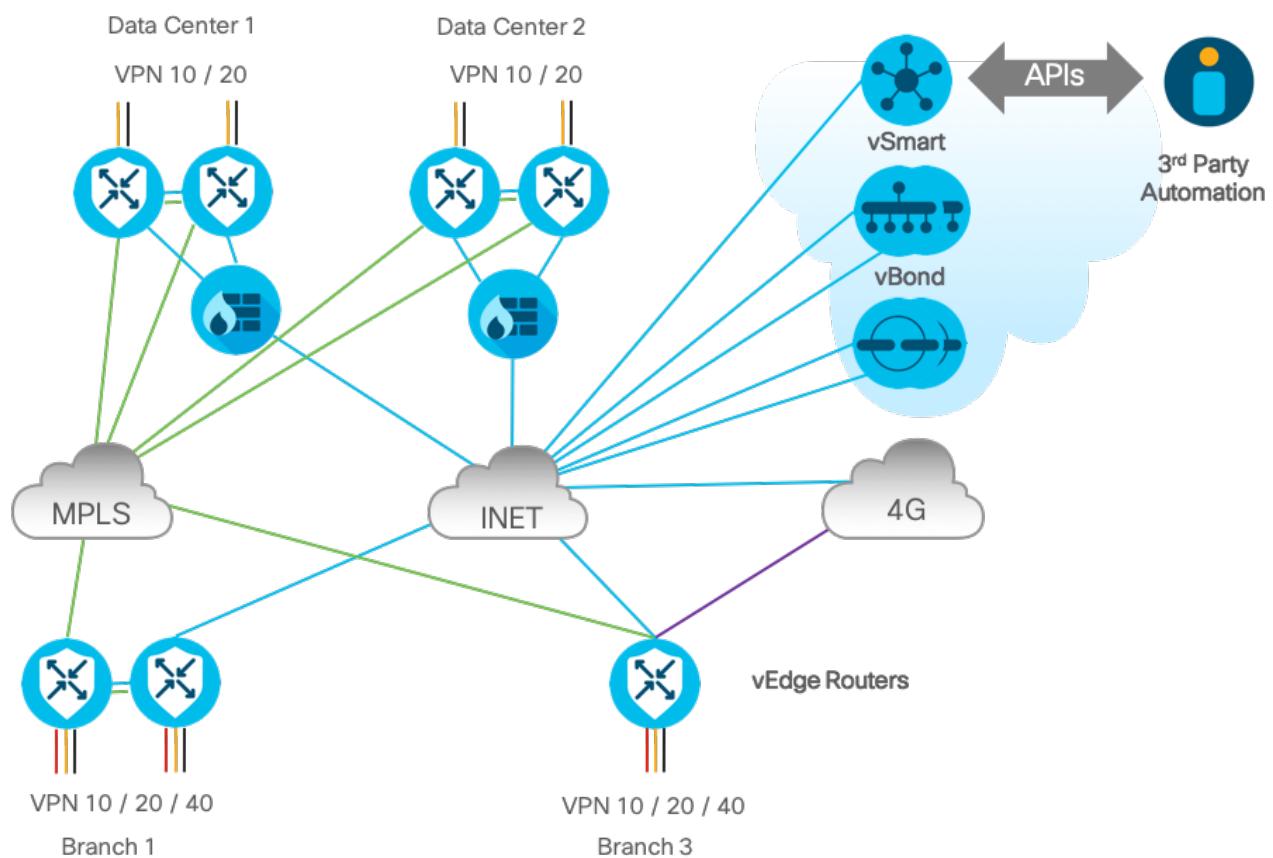
In this lab activity, you will learn how to interact with the Cisco SD WAN solutions through the use of vMANAGE REST APIs.

vMANAGE API Docs Tool, Postman and Python scripts will be used as different tools to access these APIs.

This document provides a step-by-step guide for two labs:

- 1- Use Case 1: Using vMANAGE Rest APIs, reconfigure a template associated to a particular device, re-apply it and verify the changes.
- 2- Use Case 2: Using vMANAGE Rest APIs, configure a brand-new policy in order to switch traffic between DC and a particular branch from Internet to MPLS VPN.

Network Diagram



IP Address Plan

Note: This is just for your reference (if you are using traceroute these tables will be helpful). Since we will only use DC vEDGE routers and Branch-3 cEDGE Router all other routers in the topology are excluded from these tables. Static routing, BGP and other protocols are configured and out of the scope of this lab and document.

Management IP Address Table

HOSTNAME	Management IP ADDRESS
DC1-VEDGE1	198.18.3.100
DC1-VEDGE2	198.18.3.101
DC2-VEDGE1	198.18.3.102
DC2-VEDGE2	198.18.3.103
BR1-CEDGE1	198.18.3.104
BR1-CEDGE2	198.18.3.105
BR3-CEDGE1	198.18.3.107

Hosts available in each site (per VPN) Table

SITE	Host IP in VPN 10	Host IP in VPN 20	Host IP in VPN 40
DC1	10.1.10.10	10.1.20.10	--
DC2	10.2.10.10	10.2.20.10	--
BR1	10.3.0.10	10.3.20.10	10.3.40.10
BR3	10.5.0.10	10.5.20.10	10.5.40.10

DC1-Edge1 IP Address

NEIGHBOR DEVICE	WAN IP ADDRESS
Ge 0/1 - VPN 0 (to MPLS-PE)	100.64.0.2/30
Ge 0/2 - VPN 0 (to INET-PE)	100.64.2.26/30
Ge 0/0 - VPN 10 (to FW)	10.1.10.150/24
Ge 0/3 - VPN 20 (to FW)	10.1.20.150/24

DC1-Edge2 IP Address

NEIGHBOR DEVICE	WAN IP ADDRESS
Ge 0/1 - VPN 0 (to MPLS-PE)	100.64.0.6/30
Ge 0/2 - VPN 0 (to INET-PE)	100.64.2.30/30
Ge 0/0 - VPN 10 (to FW)	10.1.10.250/24
Ge 0/3 - VPN 20 (to FW)	10.1.20.250/24

Branch 3 IP Address

NEIGHBOR DEVICE	WAN IP ADDRESS
GigabitEthernet 2 (to MPLS-PE)	100.64.0.22/30
GigabitEthernet 3 (to INET-PE)	100.64.2.10/30
GigabitEthernet 4 (VPN 10)	10.5.10.1/24
GigabitEthernet 5 (to LTE -PE)	100.64.4.10/30
GigabitEthernet 6 (VPN 20)	10.5.20.1/24
GigabitEthernet 7 (VPN 40)	10.5.40.1/24

MPLS PE ROUTER (Emulating MPLS Network) Table

NEIGHBOR DEVICE	WAN IP ADDRESS
GigabitEthernet 1 (to DC1-VEDGE1)	100.64.0.1/30
GigabitEthernet 2 (to DC1-VEDGE2)	100.64.0.5/30
GigabitEthernet 3 (to DC2-VEDGE1)	100.64.0.9/30
GigabitEthernet 4 (to DC2-VEDGE2)	100.64.0.13/30
GigabitEthernet 5 (to BR1-VEDGE1)	100.64.0.17/30
GigabitEthernet 6 (INET EDGE)	198.18.1.2/24
GigabitEthernet 7 (BR3-EDGE1)	100.64.0.21/30
GigabitEthernet 8 (BR2-EDGE1)	100.64.0.25/30

INET PE ROUTER (Emulating Internet Network) Table

NEIGHBOR DEVICE	WAN IP ADDRESS
GigabitEthernet 1 (to BR1)	100.64.2.1/30
GigabitEthernet 2 (to BR2)	100.64.2.5/30
GigabitEthernet 3 (to BR3)	100.64.2.9/30
GigabitEthernet 4 (to DC2 FTD)	100.64.2.13/30
GigabitEthernet 5 (to DC1 FTD)	100.64.2.17/29
GigabitEthernet 6 (to INET EDGE)	198.18.1.3/24

INET ROUTER (Emulating LTE Access – Internet Network) Table

NEIGHBOR DEVICE	WAN IP ADDRESS
GigabitEthernet 7 (to INET EDGE)	198.18.1.5/24
GigabitEthernet 8 (to BR1 CEDGE 2)	100.64.4.1/30
GigabitEthernet 9 (to BR2 CEDGE 1)	100.64.4.5/30
GigabitEthernet 10 (to BR3 CEDGE 1)	100.64.4.9/30

INTERNET ACCESS GATEWAY Table

NEIGHBOR DEVICE	WAN IP ADDRESS
GigabitEthernet 1 (to INET-PE / LTE-PE)	198.18.1.254/24
GigabitEthernet 2 (to INTERNET)	198.18.2.254/24

Use Case 1 / LAB1 – Change system configuration with vMANAGE APIs and feature templates

This first lab is just to show how you can use vMANAGE REST APIs to change a particular system configuration. In this case we will change the banner login and banner motd for branch 3 vedge router.

Since in our environment we have a template assigned to the device, we will change this template and reapply it.

Summary steps are:

1. vMANAGE authentication.
2. Get the template ID assigned to the device (BR3 IP ADDRESS = 10.5.0.1) and verify the banner template assigned to it.
3. Configure a new banner template.
4. Associate the brand-new banner template with the device template.
5. Re-apply the device template to branch 3 vEDGE.
6. Log in to branch 3 vEDGE device (SSH) and verify the banner login and banner MOTD.

In this lab we are going to use post-man app. In each step you will see a post-man screenshot related to the task as an additional tool for guidance.

We have created a postman collection, please feel free to download and use it.

Postman collection link:

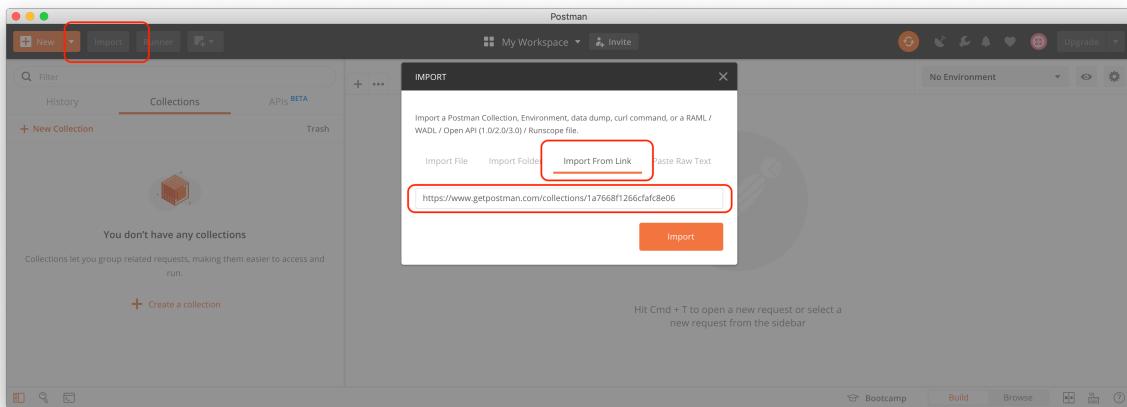
<https://www.getpostman.com/collections/12a27cd87b330ea33f90>

Quick postman review:

If you are not familiar with Postman here is a brief review on how you can import a collection:

- 1- Go to “File”, then “Import”.
- 2- Click on “Import From Link”, then paste the link:

<https://www.getpostman.com/collections/12a27cd87b330ea33f90>



Step 1: Authentication

- Login to vMANAGE using post-man application.
 - POST: https://198.18.1.10:443/j_security_check

Variable	Initial Value	Current Value
vmanage	198.18.1.10	198.18.1.10
j_username	admin	admin
j_password	admin	admin
Port	443	443

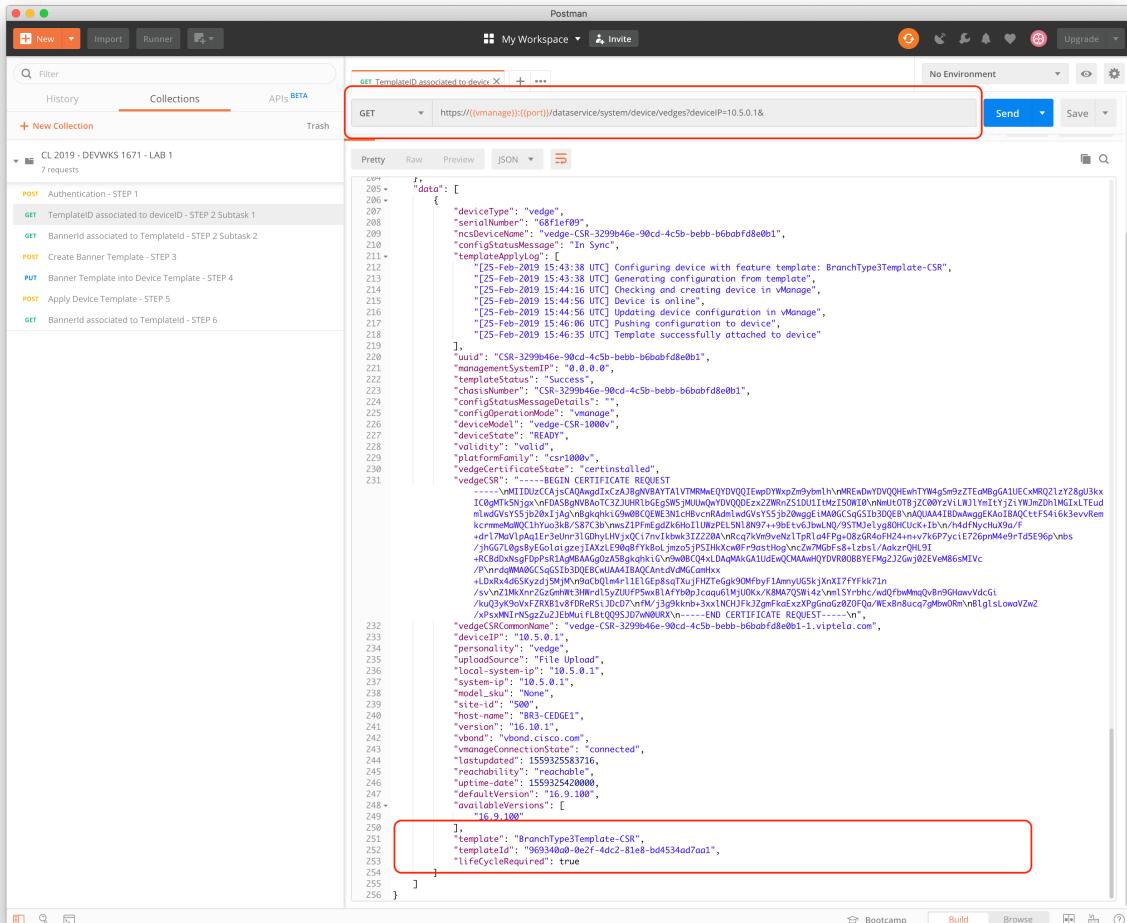
Step 2: Get the templateID associated to a deviceID

In our lab we are going to work with Branch 3 vEDGE. In this case we must have the template ID associated to the branch 3 vEDGE device ID.

- Subtask 1 – Get template ID associated to BR3 (IP ADDRESS 10.5.0.1)

- ## ○ GET

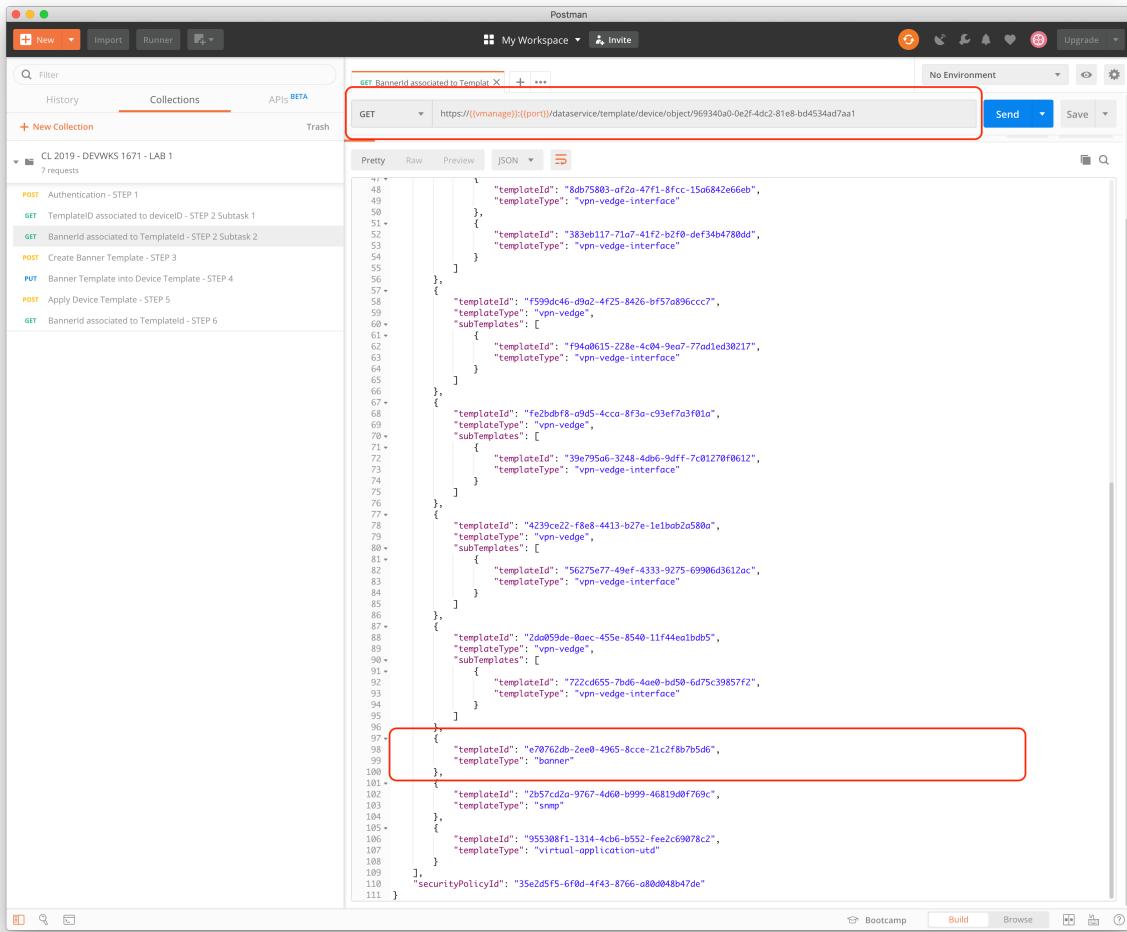
https://198.18.1.10:443/dataservice/system/device/vedges?deviceIP=1_0.5.0.1&



- Subtask 2 – Identify the banner template ID associated with the general template ID of the BR3-vEDGE

○ GEI

<https://198.18.1.10:443/dataservice/template/device/object/969340a0-0e2f-4dc2-81e8-bd4534ad7aa1>



Step 3: Create BANNER template

- Create a new banner template
 - POST <https://198.18.1.10/dataservice/template/feature/>

```
{
  "templateName": "CL2019_LAB1",
  "templateDescription": "CL2019 - Change Login banner and MOTD with APIs",
  "templateType": "banner",
  "templateMinVersion": "15.0.0",
  "templateDefinition": {
    "login": {
      "vipObjectType": "object",
      "vipType": "constant",
      "vipValue": "WARNING - Authorized personnel only",
      "vipVariableName": "banner_login"
    },
    "motd": {
      "vipObjectType": "object",
      "vipType": "constant",
      "vipValue": "WELCOME CL2019 USERS",
      "securityPolicyId": "35e2d5f5-6f0d-4f43-8766-a80d048b47de"
    }
  }
}
```

```

        "vipVariableName": "banner_motd"
    }
},
"transitionInProgress": true,
"viewMode": "add",
"deviceType": [
    "vedge-CSR-1000v"
],
"deviceModels": [
{
    "name": "vedge-CSR-1000v",
    "displayName": "CSR1000v",
    "deviceType": "vedge",
    "isCliSupported": false,
    "isCiscoDeviceModel": true
}
],
"templateUrl": "/app/configuration/template/feature/templates/banner-15.0.0.html",
"factoryDefault": false
}

```

Postman

POST Create Banner Template - STEP X

POST https://(vmanage):(port)/dataservice/template/feature/

Body (JSON application/json)

```

{
    "templateName": "CL2019_LAB1",
    "templateType": "CL2019 - Change Login banner and MOTD with APIs",
    "templateVersion": "banner",
    "templateInVersion": "15.0.0",
    "templateDefinition": {
        "login": {
            "vipObjectType": "object",
            "vipType": "constant",
            "vipValue": "WARNING : Authorized personnel only",
            "vipVariableName": "banner_login"
        },
        "motd": {
            "vipObjectType": "object",
            "vipType": "constant",
            "vipValue": "WELCOME CL2019 USERS",
            "vipVariableName": "banner_motd"
        }
    },
    "transitionInProgress": true,
    "viewMode": "add",
    "deviceType": [
        "vedge-CSR-1000v"
    ],
    "deviceModels": [
    ],
    "templateUrl": "/app/configuration/template/feature/templates/banner-15.0.0.html",
    "factoryDefault": false
}

```

Body (Pretty, Raw, Preview, JSON)

Response: Status: 200 OK Time: 871 ms Size: 391 B

1: {
2: "templateId": "a80c996d-4061-44ea-9b90-8b124ff22343"
3: }

Step 4: Associate the new banner template with the device template

- Subtask 1 – Reference the new banner template in the device template associated with BR3-vEDGE
 - PUT <https://198.18.1.10/dataservice/template/device/969340a0-0e2f-4dc2-81e8-bd4534ad7aa1>
 - BODY:

```
{  
    "templateId": "969340a0-0e2f-4dc2-81e8-bd4534ad7aa1",  
    "templateName": "BranchType3Template-CSR",  
    "templateDescription": "Branch Type 3 Template for CSR Routers",  
    "deviceType": "vedge-CSR-1000v",  
    "configType": "template",  
    "factoryDefault": false,  
    "policyId": "f73b285f-72eb-4f6d-865f-eae0e453bd8e",  
    "featureTemplateUidRange": [],  
    "connectionPreferenceRequired": true,  
    "connectionPreference": true,  
    "generalTemplates": [  
        {  
            "templateId": "3b30e089-2e26-44f1-b5b2-ac44f3f4279e",  
            "templateType": "aaa"  
        },  
        {  
            "templateId": "20d77367-06c8-4531-bad7-7e507b0c5829",  
            "templateType": "bfd-vedge"  
        },  
        {  
            "templateId": "7adf5770-deff-4472-9355-b4080b4594bd",  
            "templateType": "omp-vedge"  
        },  
        {  
            "templateId": "486d419f-4e6c-44a5-a6fb-7b5ccf94ff90",  
            "templateType": "security-vedge"  
        },  
        {  
            "templateId": "c70e876a-9ade-4e12-95e2-95fdbd4691dbe",  
            "templateType": "system-vedge",  
            "subTemplates": [  
                {  
                    "templateId": "edf3d309-91d4-45be-98d9-cfd57a05a479",  
                    "templateType": "logging"  
                }  
            ]  
        }  
    ]  
}
```

```

        ]
    },
{
    "templateId": "1bca523d-fa4c-4a43-b6d6-3250c6ec5453",
    "templateType": "vpn-vedge",
    "subTemplates": [
        {
            "templateId": "cba76e7b-584a-4b70-b625-be1537dc3568",
            "templateType": "vpn-vedge-interface"
        },
        {
            "templateId": "8db75803-af2a-47f1-8fcc-15a6842e66eb",
            "templateType": "vpn-vedge-interface"
        },
        {
            "templateId": "383eb117-71a7-41f2-b2f0-def34b4780dd",
            "templateType": "vpn-vedge-interface"
        }
    ]
},
{
    "templateId": "f599dc46-d9a2-4f25-8426-bf57a896ccc7",
    "templateType": "vpn-vedge",
    "subTemplates": [
        {
            "templateId": "f94a0615-228e-4c04-9ea7-77ad1ed30217",
            "templateType": "vpn-vedge-interface"
        }
    ]
},
{
    "templateId": "fe2bdbf8-a9d5-4cca-8f3a-c93ef7a3f01a",
    "templateType": "vpn-vedge",
    "subTemplates": [
        {
            "templateId": "39e795a6-3248-4db6-9dff-7c01270f0612",
            "templateType": "vpn-vedge-interface"
        }
    ]
},
{
    "templateId": "4239ce22-f8e8-4413-b27e-1e1bab2a580a",

```

```

    "templateType": "vpn-vedge",
    "subTemplates": [
        {
            "templateId": "56275e77-49ef-4333-9275-69906d3612ac",
            "templateType": "vpn-vedge-interface"
        }
    ],
    {
        "templateId": "2da059de-0aec-455e-8540-11f44ea1bdb5",
        "templateType": "vpn-vedge",
        "subTemplates": [
            {
                "templateId": "722cd655-7bd6-4ae0-bd50-6d75c39857f2",
                "templateType": "vpn-vedge-interface"
            }
        ]
    },
    {
        "templateId": "<TO BE CHANGED BY USER>",
        "templateType": "banner"
    },
    {
        "templateId": "2b57cd2a-9767-4d60-b999-46819d0f769c",
        "templateType": "snmp"
    },
    {
        "templateId": "955308f1-1314-4cb6-b552-fee2c69078c2",
        "templateType": "virtual-application-utd"
    }
],
"securityPolicyId": "35e2d5f5-6f0d-4f43-8766-a80d048b47de"
}

```

The screenshot shows the Postman application interface. A red box highlights the URL field which contains: `https://((vmanage)):((port))/dataservice/template/device/969340a0-0e2f-4dc2-81e8-bd4534ad7aa1`. Another red box highlights the JSON body of the request:

```

PUT
{
  "templateId": "a80c996d-4061-44ea-9b90-8b124ff22343",
  "templateType": "banner"
},
{
  "templateId": "2b57cd2a-9767-4d60-b999-46813d0f769c",
  "templateType": "smp"
},
{
  "templateId": "955308f1-1314-4cb6-b552-fee2c69078c2",
  "templateType": "virtual-application-utd"
},
{
  "securityPolicyId": "35e2d5f-6f0d-4f43-8766-a80d048b47de"
}

```

The status bar at the bottom indicates: Status: 200 OK Time: 2151 ms Size: 802 B.

Step 5: Re-apply the template associated with BR3-vEDGE

- Subtask 1 – Prior to applying the template, log in to the BR3-vEDGE (198.18.3.107) router and verify what is the login and MOTD banner.

The screenshot shows an SSH session window titled '2. ssh'. It displays the command: `MARCEGAR-M-V2C0:~ marcegar$ ssh admin@br3-e1`. The response shows the password prompt: `"CisPassword"`. Below the prompt, the router's prompt is shown: `BR3-CEDGE1#`.

- Subtask 2 – Apply the template.
 - POST
<https://198.18.1.10/dataservice/template/device/config/attachfeature>
 - BODY:

```
{
  "deviceTemplateList": [
    {
      "id": "a80c996d-4061-44ea-9b90-8b124ff22343",
      "name": "BR3-vEDGE"
    }
  ]
}
```

```

    "templateId": "969340a0-0e2f-4dc2-81e8-bd4534ad7aa1",
    "device": [
        {
            "csv-status": "complete",
            "csv-deviceId": "CSR-3299b46e-90cd-4c5b-bebb-
b6babfd8e0b1",
            "csv-deviceIP": "10.5.0.1",
            "csv-host-name": "BR3-CEDGE1",
            "/40/VPN40_INTERFACE/interface/if-name":
"GigabitEthernet7",
            "/40/VPN40_INTERFACE/interface/ip/address":
"10.5.40.1/24",
            "/20/vpn-
instance/ip/route/VPN20_TRAFFICGEN_DC1/prefix": "10.5.21.0/24",
            "/20/vpn-
instance/ip/route/VPN20_TRAFFICGEN_BR/prefix": "10.5.23.0/24",
            "/20/vpn-
instance/ip/route/VPN20_TRAFFICGEN_DC1/next-hop/VPN20_TRAFFIC_DC1_NH/address":
"10.5.20.65",
            "/20/vpn-
instance/ip/route/VPN20_TRAFFICGEN_BR/next-hop/VPN20_TRAFFIC_BR_NH/address":
"10.5.20.68",
            "/20/VPN20_INTERFACE/interface/if-name":
"GigabitEthernet6",
            "/20/VPN20_INTERFACE/interface/ip/address":
"10.5.20.1/24",
            "/10/vpn-instance/ip/route/VPN10-TRAFFIC-GEN-
DC1/prefix": "10.5.11.0/24",
            "/10/vpn-instance/ip/route/VPN10-TRAFFIC-GEN-
BR/prefix": "10.5.13.0/24",
            "/10/vpn-instance/ip/route/VPN10-TRAFFIC-GEN-
DC1/next-hop/VPN10_TRAFFIC_NH_DC1/address": "10.5.10.65",
            "/10/vpn-instance/ip/route/VPN10-TRAFFIC-GEN-
BR/next-hop/VPN10_TRAFFIC_NH_BR/address": "10.5.10.68",
            "/10/VPN10_INTERFACE/interface/if-name":
"GigabitEthernet5",
            "/10/VPN10_INTERFACE/interface/ip/address":
"10.5.10.1/24",
            "/512 vpn-instance/ip/route/0.0.0.0/0/next-
hop/VPN512_GW/address": "198.18.3.1",
            "/512/VPN512_INTERFACE/interface/if-name":
"GigabitEthernet1",
            "/512/VPN512_INTERFACE/interface/ip/address":
"198.18.3.107/24",
            "/0/vpn-instance/ip/route/0.0.0.0/0/next-
hop/MPLS_GW_IP_ADDR/address": "100.64.0.21",
            "/0/vpn-instance/ip/route/0.0.0.0/0/next-
hop/INET_GW_IP_ADDR/address": "100.64.2.9",
            "/0/LTE_TLOC_INTERFACE/interface/if-name":
"GigabitEthernet4",
            "/0/LTE_TLOC_INTERFACE/interface/ip/address":
"100.64.4.10/30",
            "/0/MPLS_TLOC_INTERFACE/interface/if-name":
"GigabitEthernet2",

```

```

        "/0/MPLS_TLOC_INTERFACE/interface/ip/address": "100.64.0.22/30",
        "/0/INET_TLOC_INTERFACE/interface/if-name": "GigabitEthernet3",
        "/0/INET_TLOC_INTERFACE/interface/ip/address": "100.64.2.10/30",
        "//system/host-name": "BR3-CEDGE1",
        "//system/gps-location/latitude": "37.33",
        "//system/gps-location/longitude": "-87.62",
        "//system/system-ip": "10.5.0.1",
        "//system/site-id": "500",
        "/0/vpn-instance/ip/route/0.0.0.0/0/next-hop/LTE_GW_IP_ADDR/address": "100.64.4.9",
        "csv-templateId": "969340a0-0e2f-4dc2-81e8-bd4534ad7aa1"
    }
],
"isEdited": true,
"isMasterEdited": false
}
]
}

```

Postman screenshot showing a POST request to apply a device template. The URL is https://(vmanage):(port)/dataservice/template/device/config/attachfeature. The request body is a JSON object containing a list of configuration details for a device template. The response status is 200 OK with a response time of 866 ms and a size of 412 B. The response body shows a single item with an ID.

```

POST https://(vmanage):(port)/dataservice/template/device/config/attachfeature
{
  "deviceTemplateList": [
    {
      "templatedId": "969340a0-0e2f-4dc2-81e8-bd4534ad7aa1",
      "device": [
        {
          "csv-status": "complete",
          "csv-deviceId": "CSR-329b46e-90cd-4c5b-bebd-b6b0bfd8e0b1",
          "csv-deviceIP": "10.5.4.104",
          "csv-deviceName": "CSR-329b46e-90cd-4c5b-bebd-b6b0bfd8e0b1",
          "csv-devicePort": "10.5.4.104",
          "csv-deviceStatus": "10.5.4.104",
          "csv-deviceType": "CSR-329b46e-90cd-4c5b-bebd-b6b0bfd8e0b1"
        },
        {
          "name": "VLAN40_INTERFACE/interface/if-name": "GigabitEthernet2",
          "ipAddress": "10.5.4.104/24",
          "macAddress": "00:0C:29:00:00:00",
          "optionalDefaultIP": "10.5.4.104/24"
        },
        {
          "name": "VLAN20_INTERFACE/interface/if-name": "GigabitEthernet6",
          "ipAddress": "10.5.20.1/24",
          "macAddress": "00:0C:29:00:00:06",
          "optionalDefaultIP": "10.5.20.1/24"
        },
        {
          "name": "VLAN10_INTERFACE/interface/if-name": "GigabitEthernet4",
          "ipAddress": "10.5.10.1/24",
          "macAddress": "00:0C:29:00:00:04",
          "optionalDefaultIP": "10.5.10.1/24"
        },
        {
          "name": "VLAN12_INTERFACE/interface/if-name": "GigabitEthernet1",
          "ipAddress": "198.18.128.1",
          "macAddress": "00:0C:29:00:00:01",
          "optionalDefaultIP": "198.18.128.1"
        },
        {
          "name": "VLAN512_INTERFACE/interface/if-name": "GigabitEthernet1",
          "ipAddress": "198.18.134.107/18",
          "macAddress": "00:0C:29:00:00:01",
          "optionalDefaultIP": "198.18.134.107/18"
        },
        {
          "name": "Vpn-instance/instance/0.0.0.0/0/next-hop/LTE_GW_IP_ADDR/address": "100.64.0.21",
          "optionalDefaultIP": "100.64.0.21"
        },
        {
          "name": "Vpn-instance/instance/0.0.0.0/0/next-hop/LTE_OPTIONAL_DEFAULT_IP_ADDRESS/address": "100.64.0.9",
          "optionalDefaultIP": "100.64.0.9"
        },
        {
          "name": "Vpn-instance/instance/0.0.0.0/0/next-hop/LTE_OPTIONAL_GW_IP_ADDRESS/address": "100.64.4.9",
          "optionalDefaultIP": "100.64.4.9"
        },
        {
          "name": "Vpn-instance/instance/0.0.0.0/0/next-hop/LTE_OPTIONAL_GW_IP_ADDRESS/address": "100.64.4.10/30",
          "optionalDefaultIP": "100.64.4.10/30"
        },
        {
          "name": "Vpn-instance/instance/0.0.0.0/0/next-hop/LTE_OPTIONAL_GW_IP_ADDRESS/address": "100.64.4.22/30",
          "optionalDefaultIP": "100.64.4.22/30"
        },
        {
          "name": "Vpn-instance/instance/0.0.0.0/0/next-hop/LTE_OPTIONAL_GW_IP_ADDRESS/address": "100.64.2.10/30",
          "optionalDefaultIP": "100.64.2.10/30"
        },
        {
          "name": "system/host-name": "BR3-CEDGE1",
          "value": "BR3-CEDGE1"
        },
        {
          "name": "system/gps-location/latitude": "37.33",
          "value": "37.33"
        },
        {
          "name": "system/gps-location/longitude": "-87.62",
          "value": "-87.62"
        },
        {
          "name": "system/system-ip": "10.5.0.1",
          "value": "10.5.0.1"
        },
        {
          "name": "system/site-id": "500",
          "value": "500"
        },
        {
          "name": "csv-templateId": "969340a0-0e2f-4dc2-81e8-bd4534ad7aa1",
          "value": "969340a0-0e2f-4dc2-81e8-bd4534ad7aa1"
        }
      ]
    }
  ]
}

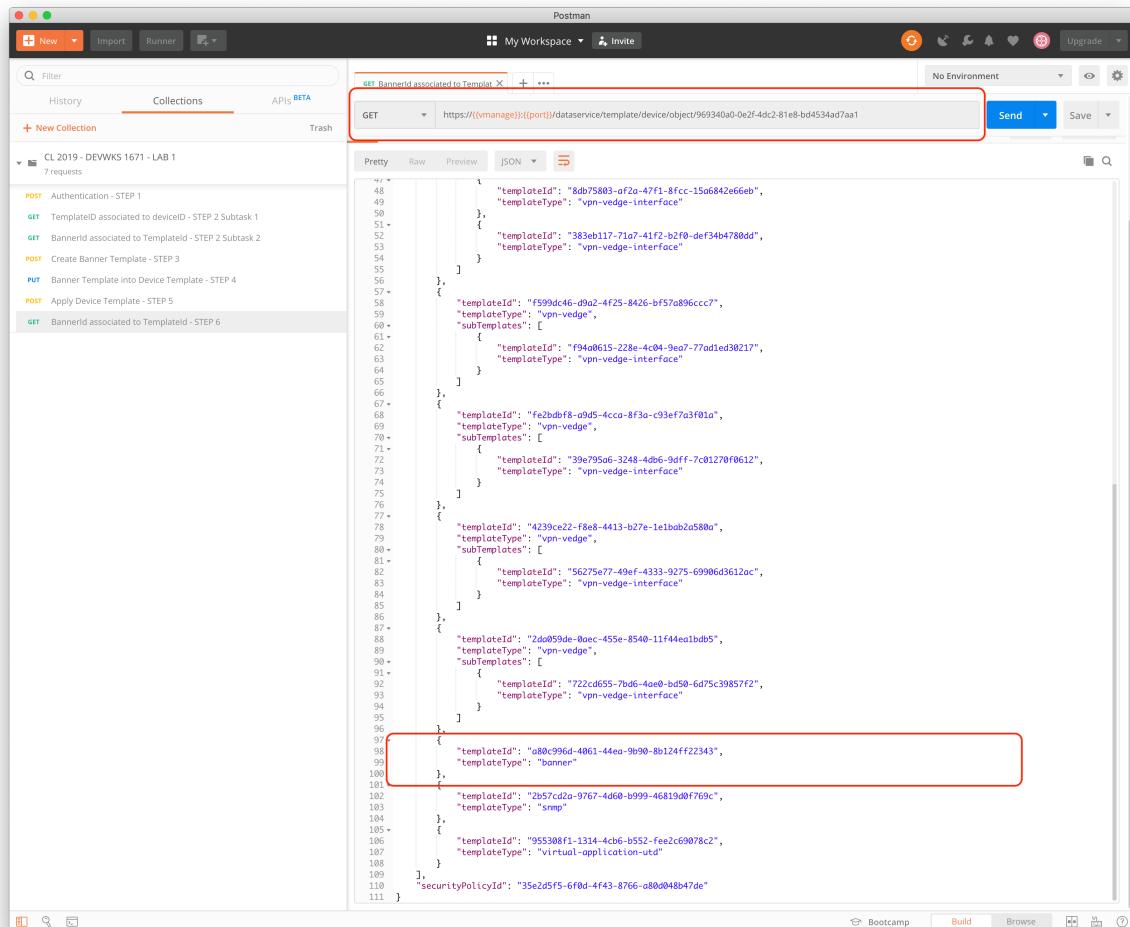
```

Step 6: Job status check

- Subtask 1 – Check the status of the job started in step 5. **Notice that you would need the job id from step 5.** Copy the job id and pasted at the end of the URI.
 - GET
[https://198.18.1.10:443/dataservice/device/action/status/"TO_BE_CHANGED_BY_USER"](https://198.18.1.10:443/dataservice/device/action/status/)
 - Perform this step (GET) until the job is done (status: success). Once the job is done proceed to the next step.

Step 7: Verification

- Subtask 1 – Verify device template (should be configured with the new banner template ID).
 - GET
<https://198.18.1.10:443/dataservice/template/device/object/969340a0-0e2f-4dc2-81e8-bd4534ad7aa1>



```
POST Authentication - STEP 1
GET TemplateID associated to deviceld - STEP 2 Subtask 1
GET BannerId associated to TemplateId - STEP 2 Subtask 2
POST Create Banner Template - STEP 3
PUT Banner Template into Device Template - STEP 4
POST Apply Device Template - STEP 5
GET BannerId associated to TemplateId - STEP 6
```

GET https://(vmanage):(port)/dataservice/template/device/object/969340a0-0e2f-4dc2-81e8-bd4534ad7aa1

Pretty Raw Preview JSON

```
[{"id": "969340a0-0e2f-4dc2-81e8-bd4534ad7aa1", "name": "New Banner Template", "description": "A new banner template created for the lab.", "templateId": "8db75803-df2a-47f1-8fcc-15a6842e66eb", "templateType": "vpn-vedge-interface"}, {"id": "383eb117-71a7-41f2-b2f0-def34b4780dd", "name": "Default Banner", "description": "The default banner template.", "templateId": "383eb117-71a7-41f2-b2f0-def34b4780dd", "templateType": "vpn-vedge-interface"}, {"id": "f94a0615-228e-4c04-9ea7-770d1ed30217", "name": "Custom Banner", "description": "A custom banner template.", "templateId": "f94a0615-228e-4c04-9ea7-770d1ed30217", "templateType": "vpn-vedge-interface"}, {"id": "fe2bdbf8-0d5d-4cca-8f3a-c93ef7a3f01a", "name": "Blank Banner", "description": "A blank banner template.", "templateId": "fe2bdbf8-0d5d-4cca-8f3a-c93ef7a3f01a", "templateType": "vpn-vedge"}, {"id": "39e795a6-3248-4db6-9df7-7c01270f0612", "name": "Standard Banner", "description": "A standard banner template.", "templateId": "39e795a6-3248-4db6-9df7-7c01270f0612", "templateType": "vpn-vedge-interface"}, {"id": "4239ce22-f8e8-4413-b27e-1e1bab2a580a", "name": "Advanced Banner", "description": "An advanced banner template.", "templateId": "4239ce22-f8e8-4413-b27e-1e1bab2a580a", "templateType": "vpn-vedge"}, {"id": "56275e77-49ef-4333-9275-69906d3612ac", "name": "Enterprise Banner", "description": "An enterprise banner template.", "templateId": "56275e77-49ef-4333-9275-69906d3612ac", "templateType": "vpn-vedge-interface"}, {"id": "2de093de-8aec-455e-8540-11f44ea1bdb5", "name": "Basic Banner", "description": "A basic banner template.", "templateId": "2de093de-8aec-455e-8540-11f44ea1bdb5", "templateType": "vpn-vedge"}, {"id": "722cd655-7bd6-4ae0-bd50-6d75c39857f2", "name": "Minimal Banner", "description": "A minimal banner template.", "templateId": "722cd655-7bd6-4ae0-bd50-6d75c39857f2", "templateType": "vpn-vedge-interface"}, {"id": "a80c996d-4061-44ea-9b90-8b124ff22343", "name": "Custom Banner", "description": "A custom banner template.", "templateId": "a80c996d-4061-44ea-9b90-8b124ff22343", "templateType": "banner"}, {"id": "2b57cd2a-9767-4d60-b999-46819d0f769c", "name": "Standard Banner", "description": "A standard banner template.", "templateId": "2b57cd2a-9767-4d60-b999-46819d0f769c", "templateType": "srmp"}, {"id": "955308f1-1314-4cb6-b552-fee2c69078c2", "name": "Virtual Application", "description": "A virtual application banner template.", "templateId": "955308f1-1314-4cb6-b552-fee2c69078c2", "templateType": "virtual-application-utd"}, {"id": "35e2d5f5-6f0d-4f43-8766-a80d048b47de", "name": "Enterprise Banner", "description": "An enterprise banner template.", "templateId": "35e2d5f5-6f0d-4f43-8766-a80d048b47de", "templateType": "virtual-application-utd"}]
```

- Subtask 2 – Login to BR3-vEDGE (198.18.3.107) and verify locally the new banner configuration.

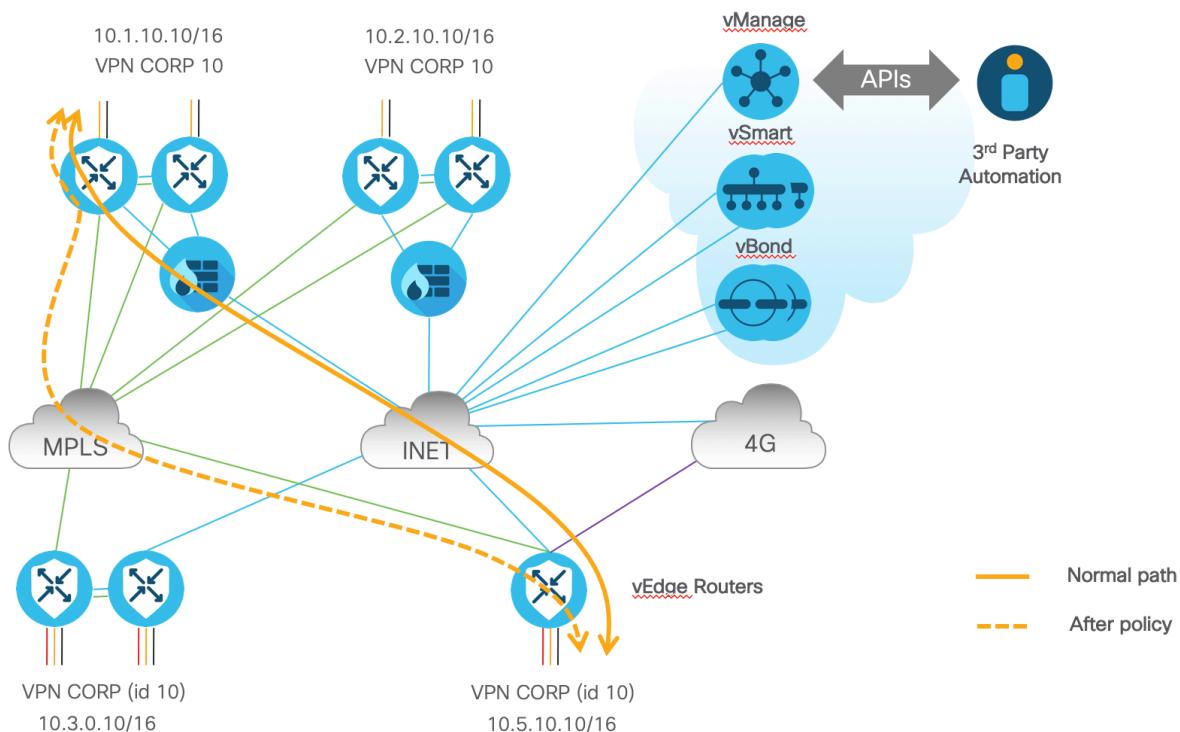
The terminal window has a dark background and light-colored text. At the top, it says "1. ssh". Below that, the command "MARCEGAR-M-V2C0:~ marcegar\$ ssh admin@198.18.134.107" is shown. A password prompt follows: "'WARNING - Authorized personnel only" Password:'. The response "WELCOME CL2019 USERS" is displayed. Finally, the prompt "BR3-CEDGE1#" is shown.

```
MARCEGAR-M-V2C0:~ marcegar$ ssh admin@198.18.134.107
"WARNING - Authorized personnel only" Password:
"WELCOME CL2019 USERS"
BR3-CEDGE1#
```

Use Case 2 / LAB 2 - Policies

Policy is used to influence the flow of data traffic among the vEdge routers in the overlay network. To implement enterprise-specific traffic control requirements, you create basic policies, and you deploy advanced features of the Cisco SD-WAN (Viptela) software that are activated by means of the policy configuration infrastructure, the policies apply either to control plane or data plane traffic, and they are configured either centrally (on vMANAGE controllers) or locally (on vEdge routers). In our scenario we'll use centralized policy.

The following figure illustrates how influences on the traffic path using policies:



Step 1: Login into vManage and validate the Policies

- Subtask 1: Log into the SD-WAN User Interface
 - Lunch your Chrome web browser and enter the following URL: <https://198.18.1.10/>. If you see a Certificate Error similar to the following, please accept it.



Your connection is not private

Attackers might be trying to steal your information from **198.18.1.10** (for example, passwords, messages, or credit cards). [Learn more](#)

NET::ERR_CERT_AUTHORITY_INVALID

Help improve Safe Browsing by sending some [system information and page content](#) to Google.
[Privacy policy](#)

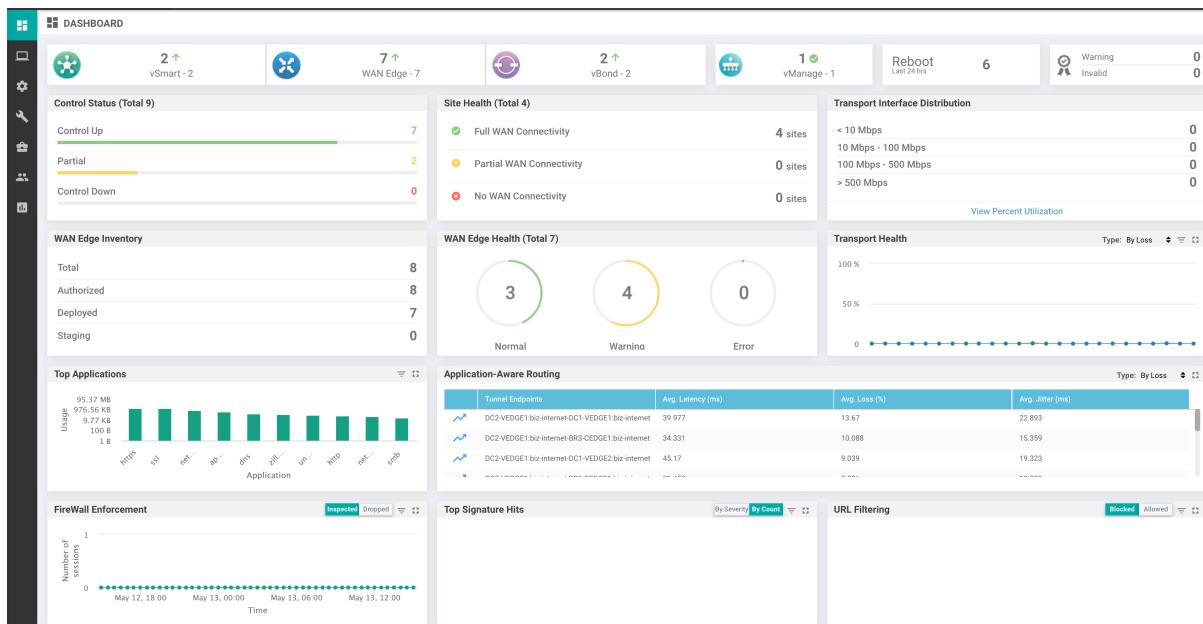
[Advanced](#)

[Back to safety](#)

- You can log into the SD-WAN solution by entering the following:
 - User: **admin**
 - Password: **admin**



- Once you have logged into the SD-WAN UI, you should see the following screen:



- Subtask 2: Validate Policies Created
 - To validate the policies created, from SD-WAN UI browse to **Configuration > Policies**. You should see all the policies created under Centralized Policy and Localized Policy, keep in mind that we'll work with Centralized policies

Configuration Policies							Custom Options
	Centralized Policy	Localized Policy					Total Rows: 8
Configuration		Search Options					
Devices	Description	Type	Activated	Updated By	Policy Version	Last Updated	
BFD/IPSec based Hub-n-Spoke Topo...	UI Policy Builder	false	admin	122820177235740495	31 Dec 2017 9:54:51 AM -05	...	
Multi-Topology Policy	UI Policy Builder	false	admin	123020177143348286	31 Dec 2017 9:56:04 AM -05	...	
Adding FW for inter-branch commun...	UI Policy Builder	false	admin	123120177143840434	31 Dec 2017 9:58:16 AM -05	...	
Application/ACL Policy for inter-bran...	UI Policy Builder	false	admin	123120177153128841	31 Dec 2017 10:35:42 AM -05	...	
App Aware Routing Policy based on ...	UI Policy Builder	false	admin	04092018T121824907	31 Dec 2017 11:00:45 AM -05	...	
BRT group prefers DC1 and BR2 grou...	UI Policy Builder	false	admin	123120177165649593	31 Dec 2017 12:00:56 PM -05	...	
cflowd	CLI	false	admin	06212018T190131728	02 Jul 2018 9:05:07 AM -05	...	
Enforce PCI VPN to be Hub and Spok...	UI Policy Builder	false	admin	02242019T10524417	24 Feb 2019 5:53:43 AM -05	...	

Step 2: Create application aware policy

- Subtask 1: Create new policy using APIs
 - Duplicate your Chrome web browser and add “apidocs/” to the original URL: <https://198.18.1.10/apidocs/>.
 - You should be able to see all the API options available to interact with Cisco SD-WAN solution:

The screenshot shows the Cisco API Explorer interface. At the top right is a search bar labeled "api_key" and a "Explore" button. Below the header, there's a list of API endpoints categorized under "Capacity", "Utility - Logging", "Alarms - Notifications", "Diagnostics", "CloudDock-Service Chain", "Resource Pool", "Configuration Database Cluster management", "Monitoring-CloudDockCluster", "CloudDock-Cluster", "Administration - Tenant", "CloudDock-Attach", "SSH", "Tenant Management", "Tenant Status", "Utility - Log files", and "Device Actions". Each endpoint has links for "Show/Hide", "List Operations", "Expand Operations", and "Raw".

- Then locate “Configuration - Policy AppRoute Definition Builder” and expand it doing click on Show/Hide to see all the options.

Configuration - Policy AppRoute Definition Builder			Show/Hide	List Operations	Expand Operations	Raw
GET	/template/policy/definition/approute/{id}				Get policy definition	
PUT	/template/policy/definition/approute/{id}				Edit policy definitions	
DELETE	/template/policy/definition/approute/{id}				Edit policy definitions	
GET	/template/policy/definition/approute				Get policy definitions	
POST	/template/policy/definition/approute				Edit policy definitions	
GET	/template/policy/definition/approute/preview/{id}				Preview policy definition	
POST	/template/policy/definition/approute/preview				Preview policy definition	

- Use the POST “**/template/policy/definition/approute**” to create an Application Aware Routing Policy and copy the following configuration as body.

```
{
  "description": "BR3 prefixes preferred path - MPLS",
  "name": "APP_AWARE_BR3_VPN10_ACTIVE_MPLS",
  "sequences": [
    {
      "actions": [
        {
          "parameter": "mpls",
          "type": "backupSlaPreferredColor"
        }
      ],
      "match": {
        "entries": [
          {
            "field": "destinationDataPrefixList",
            "ref": "81387e80-c3b2-41d7-9804-9a58a375021c"
          }
        ]
      },
      "sequenceId": 1,
      "sequenceIpType": "ipv4",
      "sequenceName": "App Route",
      "sequenceType": "appRoute"
    }
  ],
  "type": "appRoute"
}
```

POST /template/policy/definition/approuute Edit policy definitions

Implementation Notes
Create policy definition entry

Parameters

Parameter	Value	Description	Parameter Type	Data Type
body	<pre>{ "name": "APP_AWARE_BR3_VPN10_ACTIVE_MPLS", "type": "appRoute", "description": "BR3 prefixes preferred path - MPLS", "sequences": [{ "id": "1" }] }</pre>	Definition Json	body	javax.json.JsonObject

Parameter content type: application/json

Response Messages

HTTP Status Code	Reason	Response Model
200	Success	
400	Bad request	
403	Forbidden	
500	Internal Server Error	

[Try it out!](#) [Hide Response](#)

- Then click on “**Try it out**”
 - Under Response Body you will see the definition ID.
 - Copy this value because you will use it on the next step.

Response Body

```
{
  "definitionId": "d90f0d66-c7ba-46df-95ec-180cc8b3a969"
}
```

Step 3: Create a template policy using the application aware policy (configured in step 1) and a prefix list

- Subtask 1: Locate “**Configuration - vSmart Template Policy**” and expand it doing click on **Show/Hide** to see all the options.

Configuration - vSmart Template Policy [Show/Hide](#) [List Operations](#) [Expand Operations](#) [Raw](#)

POST	/template/policy/vsmart/deactivate/{policyId}	Deactivate vsmart policy
GET	/template/policy/vsmart/connectivity/status	Check VSmart Connectivity Status
GET	/template/policy/vsmart	Template list
POST	/template/policy/vsmart	Create template
GET	/template/policy/vsmart/definition/{policyId}	Get template
PUT	/template/policy/vsmart/{policyId}	Edit template
DELETE	/template/policy/vsmart/{policyId}	Delete template
POST	/template/policy/vsmart/activate/{policyId}	Activate vsmart policy

- Choose the POST “**/template/policy/vsmart**” to create a policy using the application aware policy created on step 1.

- Copy the following configuration on the body field and change the “**definitionId**” using the value saved on the previous step.

```
{
  "isPolicyActivated": false,
  "policyDefinition": [
    {
      "assembly": [
        {
          "definitionId": "<NEW ID TO BE COMPLETED BY THE USER>",
          "entries": [
            {
              "siteLists": [
                "1baf2339-97c0-43fb-98d2-285428a61820",
                "c7f1cb18-025e-4fe0-9ac6-d8b049a16789"
              ],
              "vpnLists": [
                "67c9a7e1-e1b2-4a31-bd7c-7d0bbf9b7bbb"
              ]
            }
          ],
          "type": "appRoute"
        }
      ]
    },
    {
      "policyDescription": "Cisco Live 2019 - Use Case 2 - Policy applied to branch 3 vpn10 prefixes",
      "policyName": "CL_2019_UC_2_DC1_to_BR_3_P_VPN10",
      "policyType": "feature"
    }
  ]
}
```

POST /template/policy/vsmart Create template

Implementation Notes
Create template for given policy.

Parameters

Parameter	Value	Description	Parameter Type	Data Type
body	<pre>"policyDefinition": { "assembly": [{ "definitionId": "d90f0d66-c7ba-46df-95ec-180cc8b3a969", "type": "appRoute", "entries": [{ ... }] }] }</pre>	Policy JSON	body	javax.json.JsonObject

Parameter content type: application/json

Response Messages

HTTP Status Code	Reason	Response Model
200	Success	
400	Bad request	Model Model Schema <pre>{ "error": { "message": "...", "details": "...", "code": "..." } }</pre>
403	Forbidden	
500	Internal Server Error	Model Model Schema <pre>{ "error": { "message": "...", "details": "...", "code": "..." } }</pre>

[Try it out!](#) [Hide Response](#)

- Then click on “**Try it out**”.
- At this point you should be able to see the policy created under policies using the SD-WAN UI.

CONFIGURATION | POLICIES

Custom Options ▾

Centralized Policy Localized Policy

[Add Policy](#)

Search Options ▾ Total Rows: 9

Name	Description	Type	Activated	Updated By	Policy Version	Last Updated	...
StrictHub-n-Spoke	BFD/IPSec based Hub-n-Spok...	UI Policy Builder	false	admin	12282017T235740495	31 Dec 2017 9:54:51 AM -05	...
MultiTopologyPolicy	Multi-Topology Policy	UI Policy Builder	false	admin	12302017T143348286	31 Dec 2017 9:56:04 AM -05	...
MultiTopologyPlusFWInsertion	Adding FW for inter-branch co...	UI Policy Builder	false	admin	12312017T143840434	31 Dec 2017 9:58:16 AM -05	...
MultiTopologyPlusACL	Application/ACL Policy for int...	UI Policy Builder	false	admin	12312017T153128841	31 Dec 2017 10:35:42 AM -05	...
MultiTopologyPlusAppRoute	App Aware Routing Policy bas...	UI Policy Builder	false	admin	04092018T121824907	31 Dec 2017 11:00:45 AM -05	...
DCPreferencePerRegion	BR1 group prefers DC1 and B...	UI Policy Builder	false	admin	12312017T165649593	31 Dec 2017 12:00:56 PM -05	...
cflowd_policy	cflowd	CLI	false	admin	06212018T190131728	02 Jul 2018 9:05:07 AM -05	...
CL_2019_UC_2_DC1_to_BR_3...	Cisco Live 2019 - Use Case 2 -...	UI Policy Builder	false	admin	05302019T151051376	30 May 2019 10:10:51 AM -05	...
Hub-Spoke-Policy-PCI	Enforce PCI VPN to be Hub an...	UI Policy Builder	false	admin	02242019T10524417	24 Feb 2019 5:53:43 AM -05	...

- Subtask 2: Policy verification on the data path.
 - Login to DC1 EDGE1 using terminal console “**ssh admin@198.18.3.100**” the password is “**admin**”

- Traceroute from DC1 EDGE1 to BR3 and verify the change in the path. Use the following command “**traceroute vpn 10 10.5.10.10**” The trace should show the IP address of the link between the INET and BR3 (100.64.2.10)

```
Last login: Mon Oct 21 08:12:13 on ttys001
JAILEON-M-JFTB:~ jaileon$ ssh admin@198.18.3.100
Warning: Permanently added '198.18.3.100' (ECDSA) to the list of known hosts.
Cisco SD-WAN/Viptela dCloud Demo V2
admin@198.18.3.100's password:
Last login: Mon Oct 21 22:17:08 2019 from 10.16.60.145
Welcome to Viptela CLI
admin connected from 10.16.60.145 using ssh on DC1-EDGE1
DC1-EDGE1# traceroute vpn 10 10.5.10.10
Traceroute 10.5.10.10 in VPN 10
traceroute to 10.5.10.10 (10.5.10.10), 30 hops max, 60 byte packets
1 100.64.2.10 (100.64.2.10) 14.051 ms 14.694 ms 14.882 ms
2 10.5.10.10 (10.5.10.10) 14.826 ms 14.757 ms 15.002 ms
DC1-EDGE1#
```

Step 4: Activate policy

- Subtask 1: To activate the policy you must have the "policyId" value of the policy created.
 - To get this value use the option GET “**/template/policy/vsmart**” under **“Configuration - vSmart Template Policy”**.
 - Look the response body for the description of the policy configured in “Step 2”, then look for the attribute associated to “policyId” and copy it.

Response Body

```
{
  "policyId": "1c865c35-05ac-43a7-a6ee-6328138cd1c4",
  "createdBy": "admin",
  "policyType": "cli",
  "lastUpdatedOn": 1530540307108
},
{
  "policyVersion": "05302019T151051376",
  "lastUpdatedBy": "admin",
  "policyName": "CL_2019_UC_2_DC1_to_BR_3_P_VPN10",
  "policyDefinition": "{\"assembly\": [{\"definitionId\": \"d90f0d66-c7ba-46df-95ec-180cc8b3a969\", \"type\": \"appR\"}",
  "createdOn": 159229051376,
  "isPolicyActivated": false,
  "policyDescription": "Cisco Live 2019 - Use Case 2 - Policy applied to branch 3 vpn10 prefixes",
  "grid": 500,
  "policyId": "eb9d9689-6778-4af0-8fc5-e34701096101",
  "createdBy": "admin",
  "policyType": "feature",
  "lastUpdatedOn": 159229051376
},
```

- Under **“Configuration - vSmart Template Policy”** use the POST **“/template/policy/vsmart/activate/{policyId}”** option.
- Copy the value previously identified (policyId attribute) and paste into the policyId field.
- Finally, on the body field paste the following.

```
{
  "isEdited":false
}
```

POST /template/policy/vsmart/activate/{policyId} Activate vsmart policy

Implementation Notes
Activate vsmart policy for a given policy id.

Parameters

Parameter	Value	Description	Parameter Type	Data Type
policyId	eb9d9689-6778-4af0-8fc5-e34701096101	Policy Id	path	string
body	{ "IsEdited":false }	Policy JSON	body	javax.json.JsonObject

Parameter content type: application/json

Response Messages

HTTP Status Code	Reason	Response Model
200	Success	
400	Bad request	Model Model Schema <pre>{ "error": { "message": "", "details": "", "code": "" } }</pre>
403	Forbidden	
500	Internal Server Error	Model Model Schema <pre>{ "error": { "message": "", "details": "", "code": "" } }</pre>

[Try it out!](#) [Hide Response](#)

Step 5: Verification

- Subtask 1: Policy verification on vMANAGE.
 - Return to the configuration policies on vMANAGE UI and verify that the policy is activated:

CONFIGURATION | POLICIES Custom Options ▾

Centralized Policy		Localized Policy						
Add Policy				Total Rows: 9 Custom Options ▾				
<input type="text"/> Search Options ▾								
Name	Description	Type	Activated	Updated By	Policy Version	Last Updated		
StrictHub-n-Spoke	BFD/IPSec based Hub-n-Spok...	UI Policy Builder	false	admin	12282017T235740495	31 Dec 2017 9:54:51 AM -05	...	
MultiTopologyPolicy	Multi-Topology Policy	UI Policy Builder	false	admin	12302017T143348286	31 Dec 2017 9:56:04 AM -05	...	
MultiTopologyPlusFWinsertion	Adding FW for inter-branch co...	UI Policy Builder	false	admin	12312017T143840434	31 Dec 2017 9:58:16 AM -05	...	
MultiTopologyPlusACL	Application/ACL Policy for int...	UI Policy Builder	false	admin	12312017T153128841	31 Dec 2017 10:35:42 AM -05	...	
MultiTopologyPlusAppRoute	App Aware Routing Policy bas...	UI Policy Builder	false	admin	04092018T121824907	31 Dec 2017 11:00:45 AM -05	...	
DCPreferencePerRegion	BR1 group prefers DC1 and B...	UI Policy Builder	false	admin	12312017T165649593	31 Dec 2017 12:00:56 PM -05	...	
cflowd_policy	cflowd	CLI	false	admin	06212018T190131728	02 Jul 2018 9:05:07 AM -05	...	
CL_2019.UC_2_DC1_to_BR_3...	Cisco Live 2019 - Use Case 2 ...	UI Policy Builder	true	admin	05302019T151051376	30 May 2019 10:10:51 AM -05	...	
Hub-Spoke-Policy-PCI	Enforce PCI VPN to be Hub an...	UI Policy Builder	false	admin	02242019T10524417	24 Feb 2019 5:53:43 AM -05	...	

- Subtask 2: Policy verification on the data path.

- Login to DC1 EDGE1 using terminal console “**ssh admin@198.18.3.100**” the password is “**admin**”
- Traceroute from DC1 EDGE1 to BR3 and verify the change in the path. Use the following command “**traceroute vpn 10 10.5.10.10**” The trace should show you the IP address of the link between the MPLS-PE and BR3 (100.64.0.22)

```

JAILEON-M-JFTB:~ jaileon$ ssh admin@198.18.3.100
Warning: Permanently added '198.18.3.100' (ECDSA) to the list of known hosts.
Cisco SD-WAN/Viptela dCloud Demo V2
admin@198.18.3.100's password:
Last login: Mon Oct 21 23:43:31 2019 from 10.16.60.145
Welcome to Viptela CLI
admin connected from 10.16.60.145 using ssh on DC1-VEDGE1
DC1-VEDGE1# traceroute vpn 10 10.5.10.10
Traceroute 10.5.10.10 in VPN 10
traceroute to 10.5.10.10 (10.5.10.10), 30 hops max, 60 byte packets
1 100.64.0.22 (100.64.0.22) 4.181 ms 4.082 ms 4.810 ms
2 10.5.10.10 (10.5.10.10) 4.901 ms 4.808 ms 4.827 ms
DC1-VEDGE1#

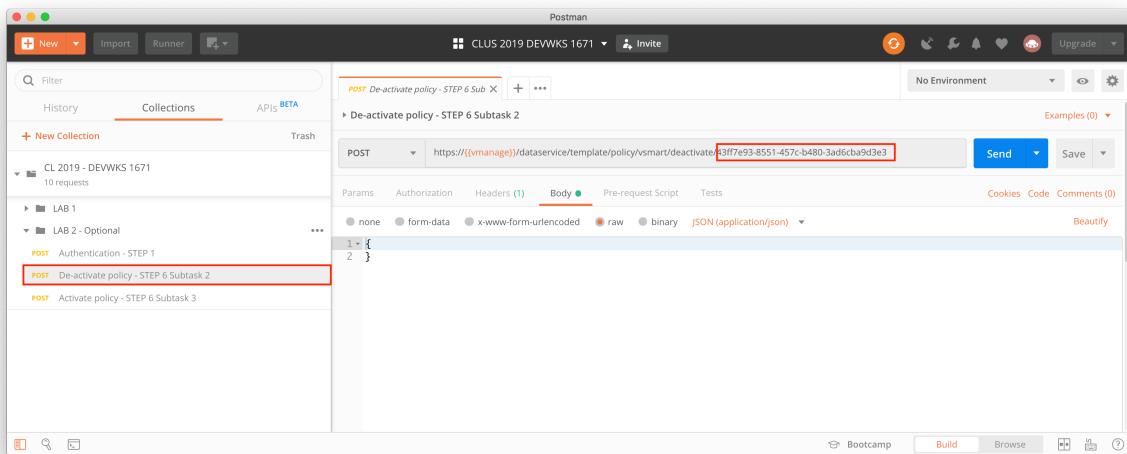
```

Step 6: Deactivate policy using Postman (OPTIONAL)

You can use postman as another tool to interact with vManage APIs.

- Subtask 1: Open Postman and use the same collection imported during LAB1, but in this case use the scripts under LAB 2 folder.
 - Execute the “**POST Login**” to get the cookie from vmanage

- Subtask 2: De-activate the policy:
 - Execute the “**POST de-activate policy**”, to proceed with this step you need change the PolicyID on the URL, the PolicyID is the same value that you used on the step 4



- Then, go to SD-WAN UI browse to **Configuration > Policies** and verify that the policy is disabled (false under Activated column).

CONFIGURATION POLICIES									Custom Options
Centralized Policy		Localized Policy							
Add Policy									Total Rows: 9
Name	Description	Type	Activated	Updated By	Policy Version	Last Updated			
StrictHub-n-Spoke	BFD/IPSec based Hub-n-Spok...	UI Policy Builder	false	admin	122820177235740495	31 Dec 2017 9:54:51 AM -05
MultiTopologyPolicy	Multi-Topology Policy	UI Policy Builder	false	admin	123020177143348286	31 Dec 2017 9:56:04 AM -05
MultiTopologyPlusFWInsertion	Adding FW for inter-branch co...	UI Policy Builder	false	admin	123120177143840434	31 Dec 2017 9:58:16 AM -05
MultiTopologyPlusACL	Application/ACL Policy for int...	UI Policy Builder	false	admin	123120177153128841	31 Dec 2017 10:35:42 AM -05
MultiTopologyPlusAppRoute	App Aware Routing Policy bas...	UI Policy Builder	false	admin	04092018T121824907	31 Dec 2017 11:00:45 AM -05
DCPreferencePerRegion	BR1 group prefers DC1 and B...	UI Policy Builder	false	admin	123120177165649593	31 Dec 2017 12:00:56 PM -05
cflowd_policy	cflowd	CLI	false	admin	06212018T190131728	02 Jul 2018 9:05:07 AM -05
CL_2019_UC_2_DC1_to_BR_3...	Cisco Live 2019 - Use Case 2 -...	UI Policy Builder	false	admin	05302019T151051376	30 May 2019 10:10:51 AM -05
Hub-Spoke-Policy-PCI	Enforce PCI VPN to be Hub an...	UI Policy Builder	false	admin	02242019T10524417	24 Feb 2019 5:53:43 AM -05

- Subtask 3: Activate the policy:

- Execute the “**POST activate policy**”, to proceed with this step you need change the PolicyID on the URL, the PolicyID is the same value that you used on the step 4

References

- Github repository:
 - <https://github.com/jaileon/DEVWKS-1671.git>
- Postman Collection:
 - <https://www.getpostman.com/collections/12a27cd87b330ea33f90>