
cisco Live!
Let's go

The bridge to possible

Automate Cisco IOS XE Device Configuration Using Terraform

Story DeWeese, Technical Marketing
@StoryDeWeese

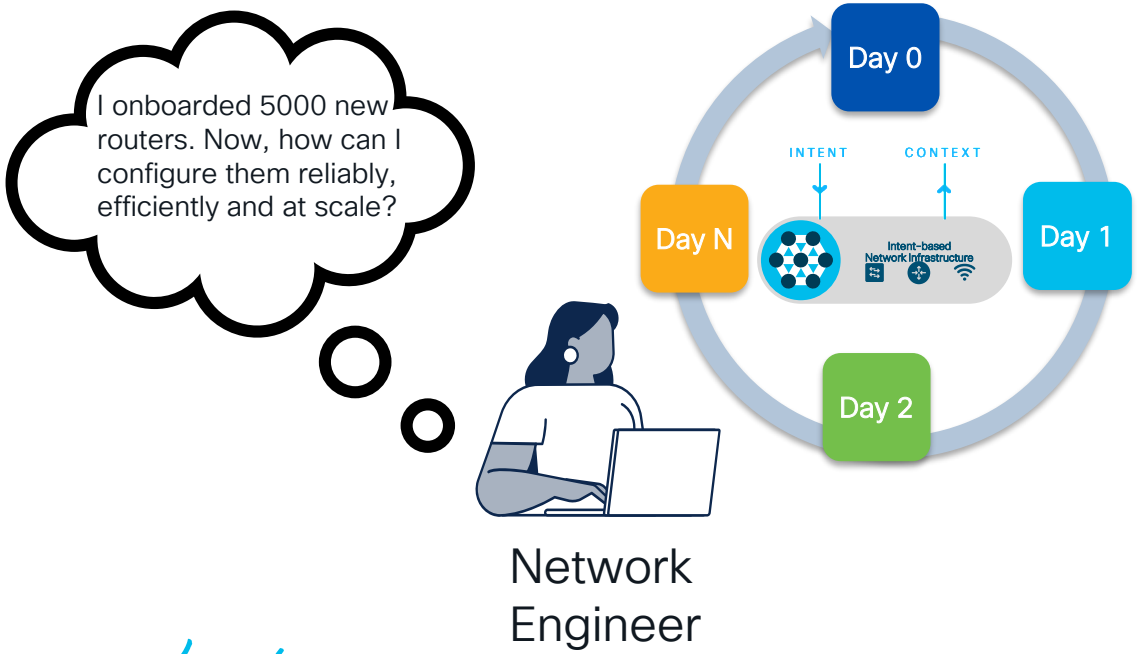
cisco Live!

DEVLIT-2083

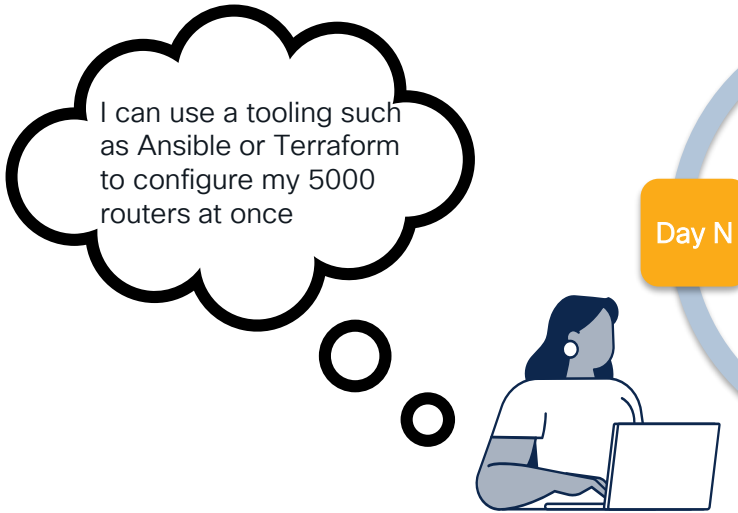
Agenda

- Terraform
- Use Cases & Demos
- Resources

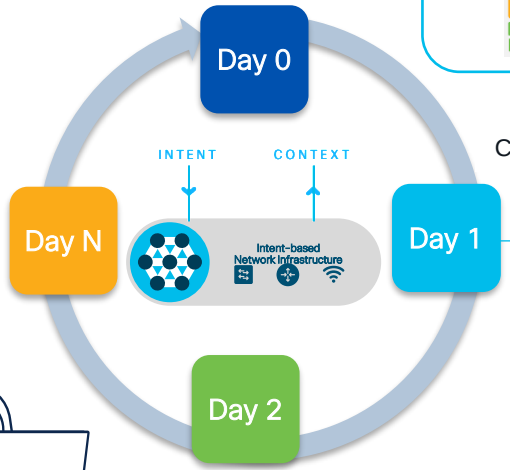
Cisco IOS XE Programmability & Automation Lifecycle



Cisco IOS XE Programmability & Automation Lifecycle



Network Engineer



Model Driven Programmability

Device Configuration

Network Configuration Protocol (NETCONF), RESTCONF, gNMI

YANG “native” Data Models, OpenConfig,

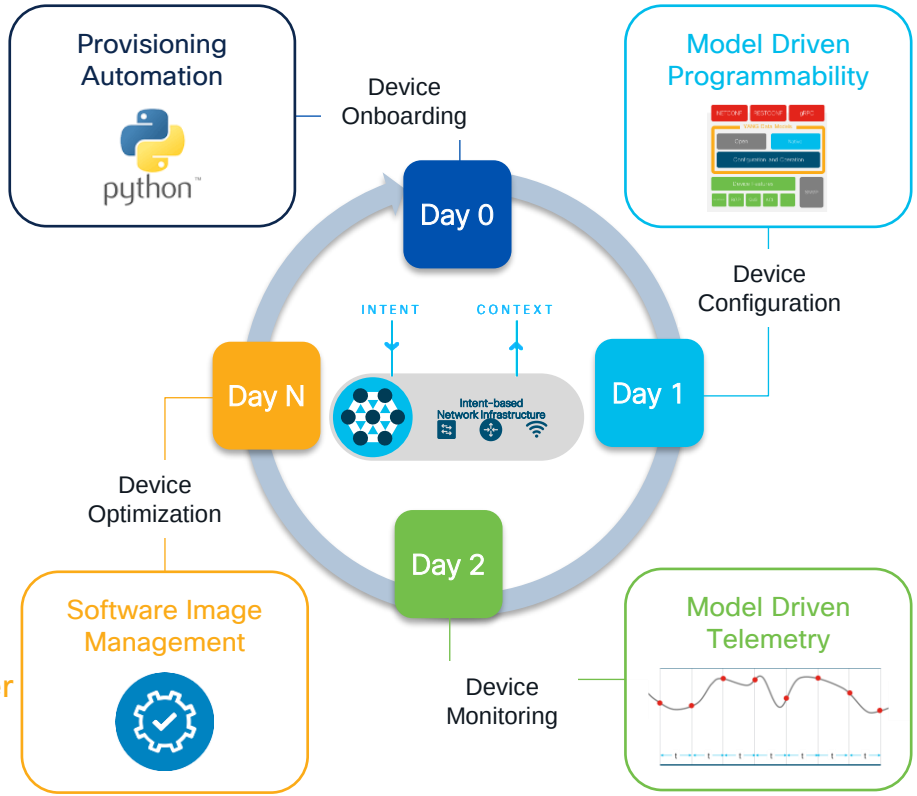
YANG Suite, Terraform, Ansible, pyATS tooling

IOS XE Programmability & Automation Lifecycle

Pre-boot Execution Environment (iPXE)

Zero Touch Provisioning

VM Automation



Network Configuration Protocol (NETCONF), RESTCONF, gNMI

YANG Data Models, OpenConfig, and YANG Suite tooling

Terraform, Ansible, pyATS

gNOI cert/os/reset proto

Guest Shell + NETCONF

CentOS 8 Python 3

Application Hosting with Docker

CLI to XML

TIG_MDT container + examples

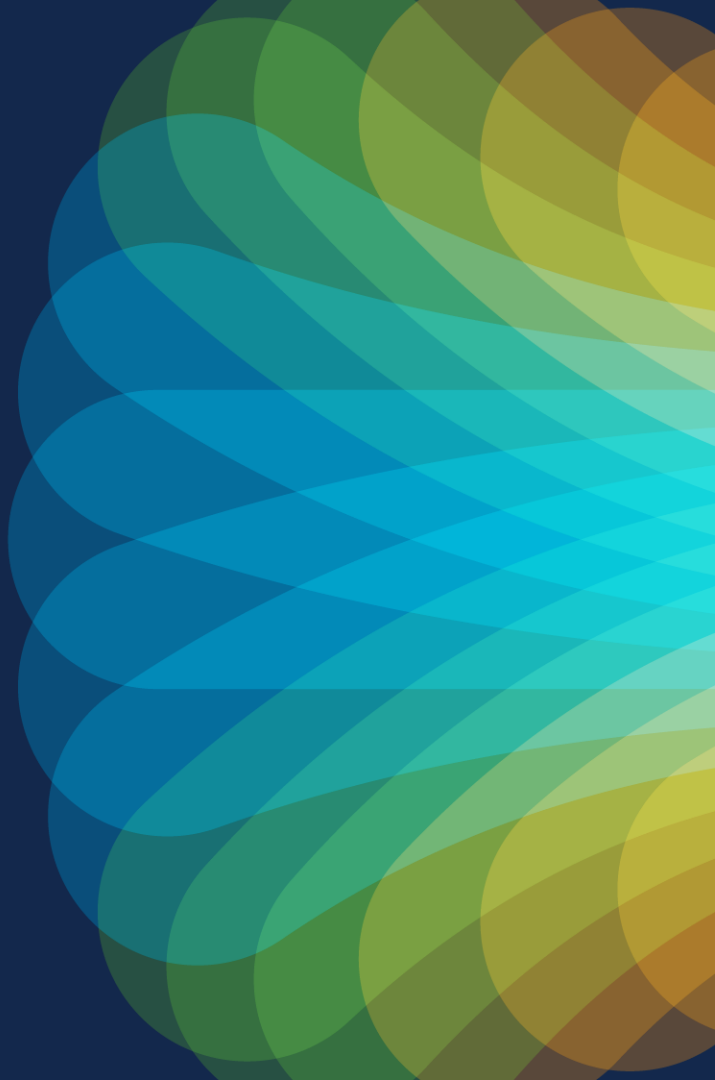
YANG On-Change support

gRPC Dial-Out + DNS + TLS

gNMI/NETCONF Dial-In

cisco Live!

Terraform

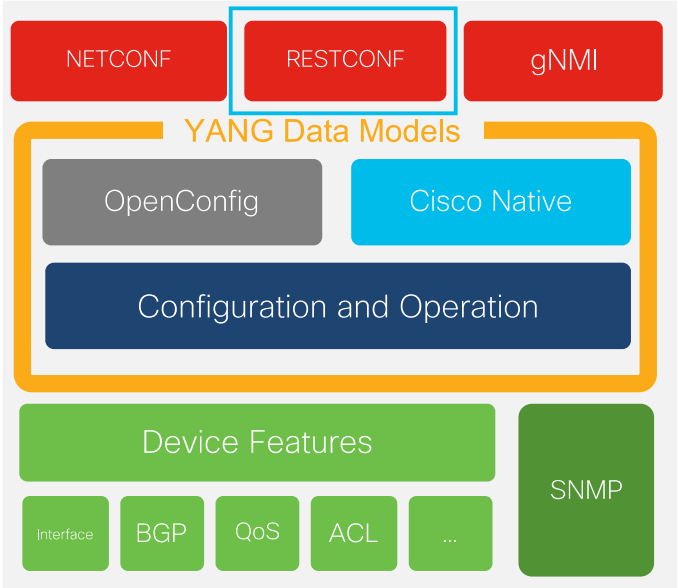


Terraform is...

Infrastructure as Code (IaC) Software Tool providing a consistent CLI workflow to manage hundreds of cloud services. Terraform codifies cloud APIs into declarative configuration files.

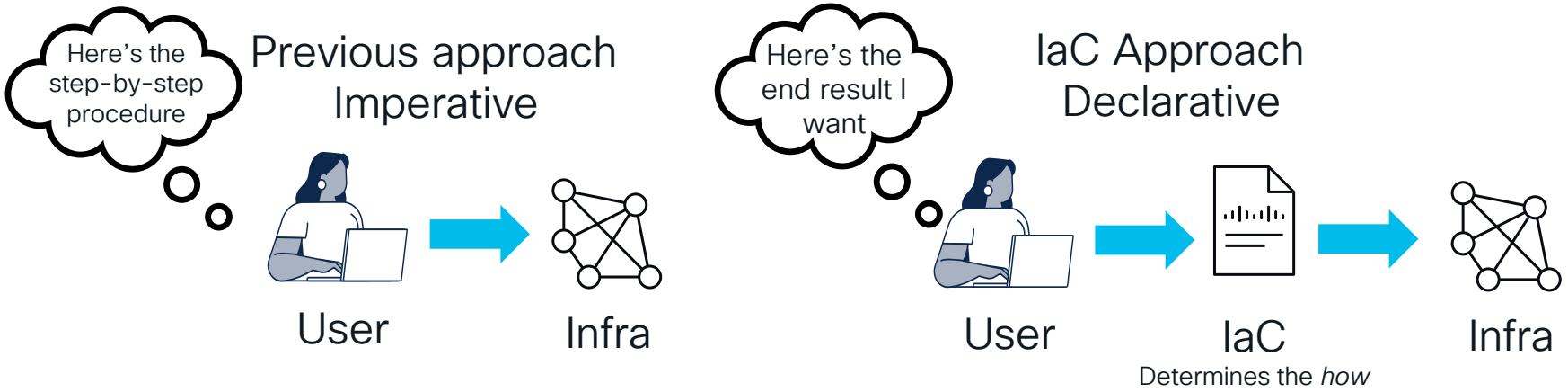
- Cloud Native Tooling circa 2014 from HashiCorp
- Agentless, single binary file
- Zero server-side dependencies

Terraform uses the RESTCONF API



What's IaC?

Infrastructure as Code (IaC) is the process of managing changes through code, rather than a manual process

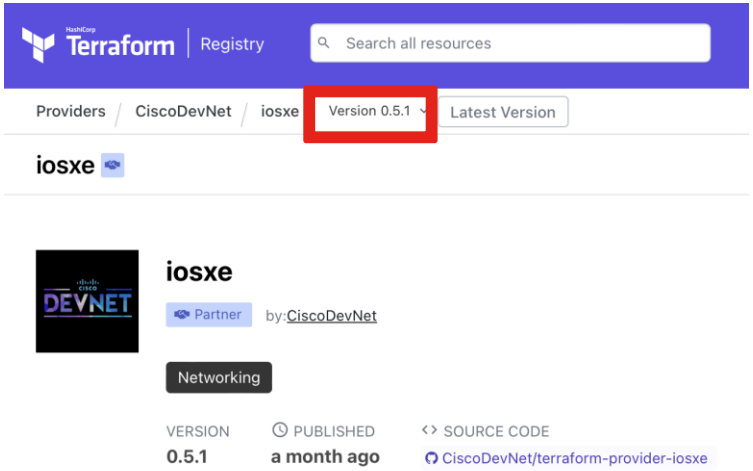


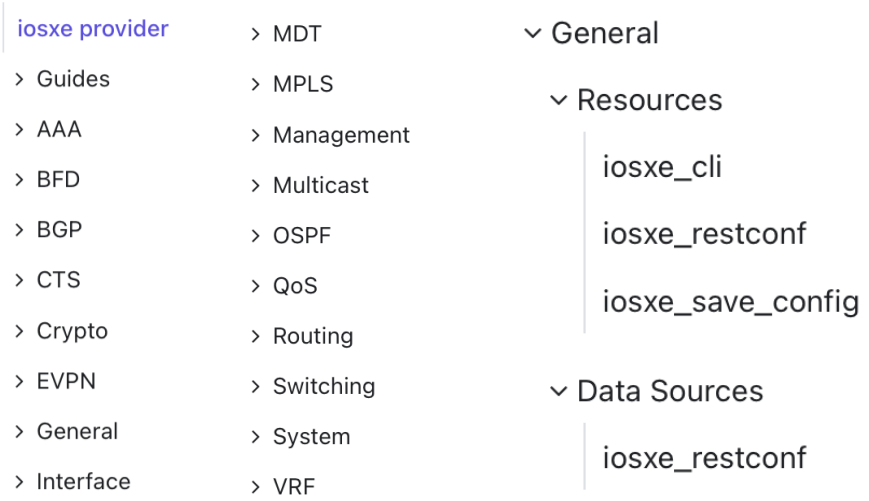
Learn more about IaC here:
https://developer.cisco.com/iac/#:~:text=Adopting%20Infrastructure%20as%20Code%20allows,data%20center%20to%20the%20edge.

IOS XE Terraform Provider

Documentation and details about the provider are available on the Hashicorp Registry
https://registry.terraform.io/providers/CiscoDevNet/iosxe/latest

Source code is in the GitHub Repository: https://github.com/CiscoDevNet/terraform-provider-iosxe/





Terraform resource utilizing the CLI RPC

https://registry.terraform.io/providers/CiscoDevNet/iosxe/latest/docs/resources/cli

iosxe_cli (Resource)

This resources is used to configure arbitrary CLI commands. This should be considered a last resort in case YANG models are not available, as it cannot read the state and therefore cannot reconcile changes.

Example Usage

```
resource "iosxe_cli" "example" {
  cli = <<-EOT
  interface Loopback123
  description configured-via-restconf-cli
  EOT
}
```

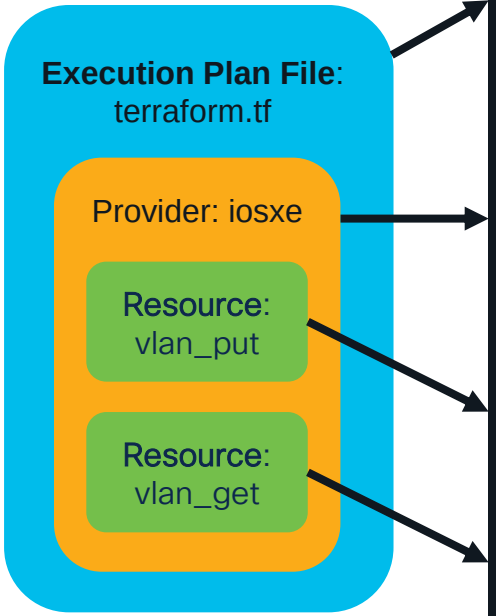
Terraform Terminology

Terraform uses an execution plan file with a provider and resource definitions

An **execution plan file** defines the provider and resources. It is written in HashiCorp Configuration Language (HCL), similar to JSON, and stored with a .tf extension

A **provider** is a plugin to make a collection of resources accessible

A **resource** (or infrastructure resource) describes one or more infrastructure objects managed by Terraform. With the IOS XE Terraform provider, resources can be considered the same as a configurable feature




```
terraform {
        required_providers {
                iosxe = {
                        source = "CiscoDevNet/iosxe"
                        version = ">=0.5.0"
                }
        }
}

provider "iosxe" {
        username = var.device_username
        password = var.device_password
        url = var.host_url
}


resource "iosxe_vlan" "example" {
        vlan_id  = 511
        name     = "Vlan511"
        shutdown = false
}

data "iosxe_vlan" "example" {
  vlan_id = 511
}
```

CLI to YANG

This new CLI addition to “show run | format” brings additional visibility into the YANG modelled configuration, either for NETCONF with XML or JSON with RESTCONF
Easily convert CLI into YANG to re-use in tooling, scripts, and automation and orchestration systems

```
show run | format netconf-xml
show run | format restconf-json
```

```
C9300#show run | format netconf-xml
<config xmlns="http://tail-f.com/ns/config/1.0">
  <native xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-native">
    <version>17.7</version>
    <memory>
      <free>
        <low-watermark>
          <processor>131752</processor>
        </low-watermark>
```

```
C9300#show run | format restconf-json
{
  "data": {
    "Cisco-IOS-XE-native:native": {
      "version": "17.7",
      "memory": {
        "free": {
          "low-watermark": {
            "processor": 131923
```

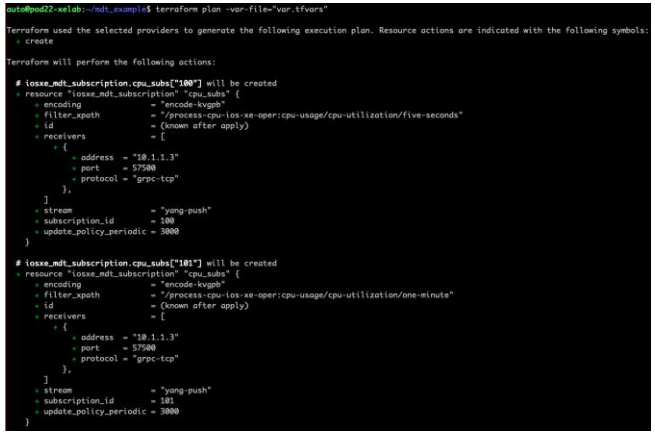
```
C9300#
C9300#show run | i netconf-yang
netconf-yang
C9300#
```

Requires `netconf-yang` Data Model Interfaces to be enabled
CLIs with corresponding native YANG and modeled in show run are returned

Use Cases & Demos

Terraform for Model Driven Telemetry

Enable gRPC Dial-Out telemetry subscriptions for the POE and Basic Device Monitoring use cases



```
auto@pod22-xelab:~/mdt_example$ terraform plan -var-file="var.tfvars"

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # iosxe_mdt_subscription.cpu_subs["100"] will be created
  + resource "iosxe_mdt_subscription" "cpu_subs" {
      + encoding          = "encode-kvgpb"
      + filter_xpath      = "/process-cpu-ios-xe-oper:cpu-usage/cpu-utilization/five-seconds"
      + id                = (known after apply)
      + receivers         = [
          + {
              + address  = "10.1.1.3"
              + port     = 57500
              + protocol = "grpc-tcp"
            },
        ]
      + stream            = "yang-push"
      + subscription_id   = 100
      + update_policy_periodic = 3000
    }

  # iosxe_mdt_subscription.cpu_subs["101"] will be created
  + resource "iosxe_mdt_subscription" "cpu_subs" {
      + encoding          = "encode-kvgpb"
      + filter_xpath      = "/process-cpu-ios-xe-oper:cpu-usage/cpu-utilization/one-minute"
      + id                = (known after apply)
      + receivers         = [
          + {
              + address  = "10.1.1.3"
              + port     = 57500
              + protocol = "grpc-tcp"
            },
        ]
      + stream            = "yang-push"
      + subscription_id   = 101
      + update_policy_periodic = 3000
    }
```

Lab Guide Steps:

1. Introduction
2. IOS XE CLI pre-req
3. Terraform Install & headers, device variables
4. Variables and config for MDT subscription
5. Terraform configuration for device monitoring
6. TF Workflow: init, plan, apply & destroy
7. Validation with TF and CLI
8. Conclusion

iosxe_mdt_subscription (Resource)

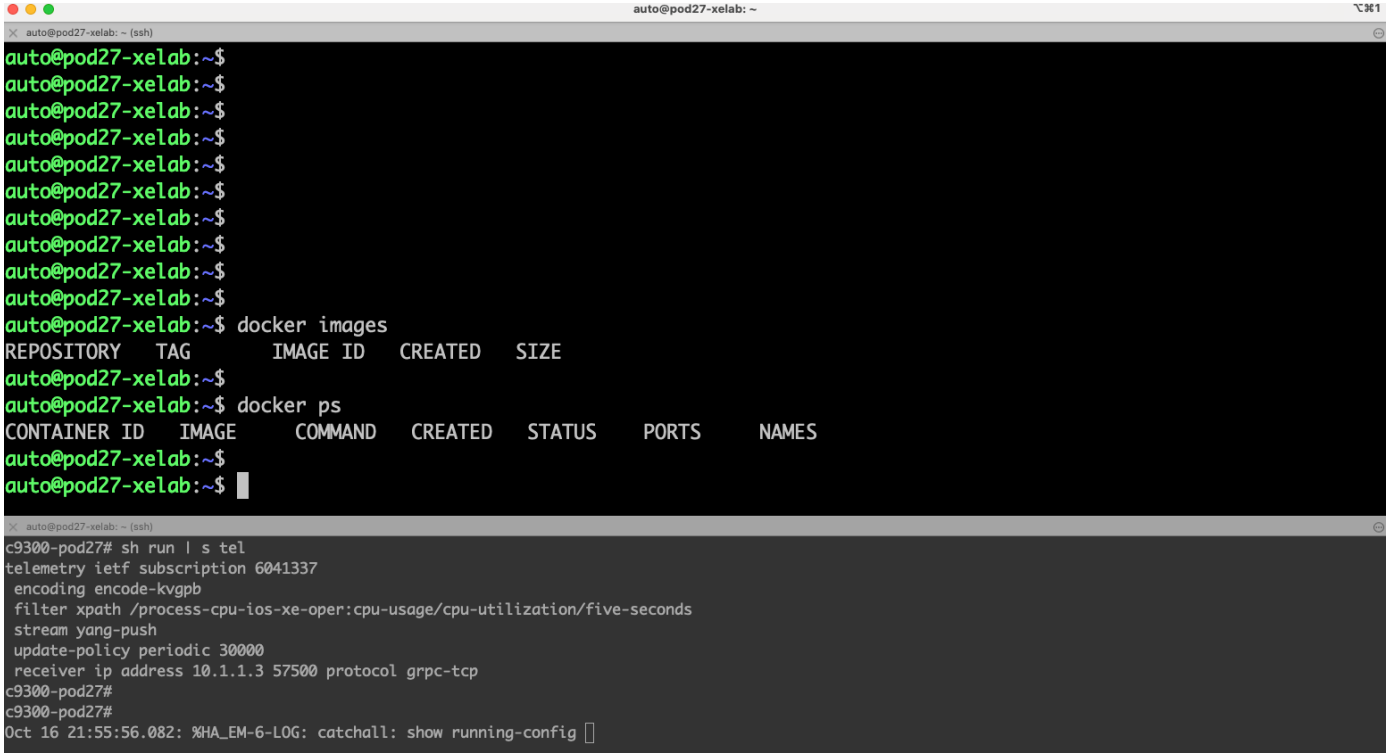
This resource can manage the MDT Subscription configuration.

Example Usage

```
resource "iosxe_mdt_subscription" "example" {
  subscription_id         = 101
  stream                  = "yang-notif-native"
  encoding                = "encode-kvgpb"
  source_vrf              = "Mgmt-vrf"
  source_address          = "1.2.3.4"
  update_policy_on_change = true
  filter_xpath            = "/ios-events-ios-xe-oper
  receivers = [
    {
      address  = "5.6.7.8"
      port     = 57600
      protocol = "grpc-tcp"
    }
  ]
}
```

https://registry.terraform.io/providers/CiscoDevNet/iosxe/latest/docs/resources/mdt_subscription
https://registry.terraform.io/providers/CiscoDevNet/iosxe/latest/docs/data-sources/mdt_subscription
https://github.com/jeremycohoe/cisco-ios-xe-panda-lab-terraform

DEMO – install docker container and use Terraform files to configure telemetry subscriptions



Use the Docker Container with the Terraform files included!

1. Ensure Docker is installed
2. Run the following commands
    1. docker pull jeremycohoe/tig_mdt
    2. docker run -ti -p 3000:3000 -p 57500:57500 jeremycohoe/tig_mdt
3. Identify the container ID
    1. docker ps

```
sdeweese@SDEWEESE-M-C20V ~ % docker ps
CONTAINER ID   IMAGE                COMMAND       CREATED          STATUS          PORTS                                                  NAMES
288fa40fd38b   jeremycohoe/tig_mdt   "/start -d"   43 seconds ago   Up 42 seconds   0.0.0.0:3000->3000/tcp, 0.0.0.0:57500->57500/tcp   inspiring_gould
```

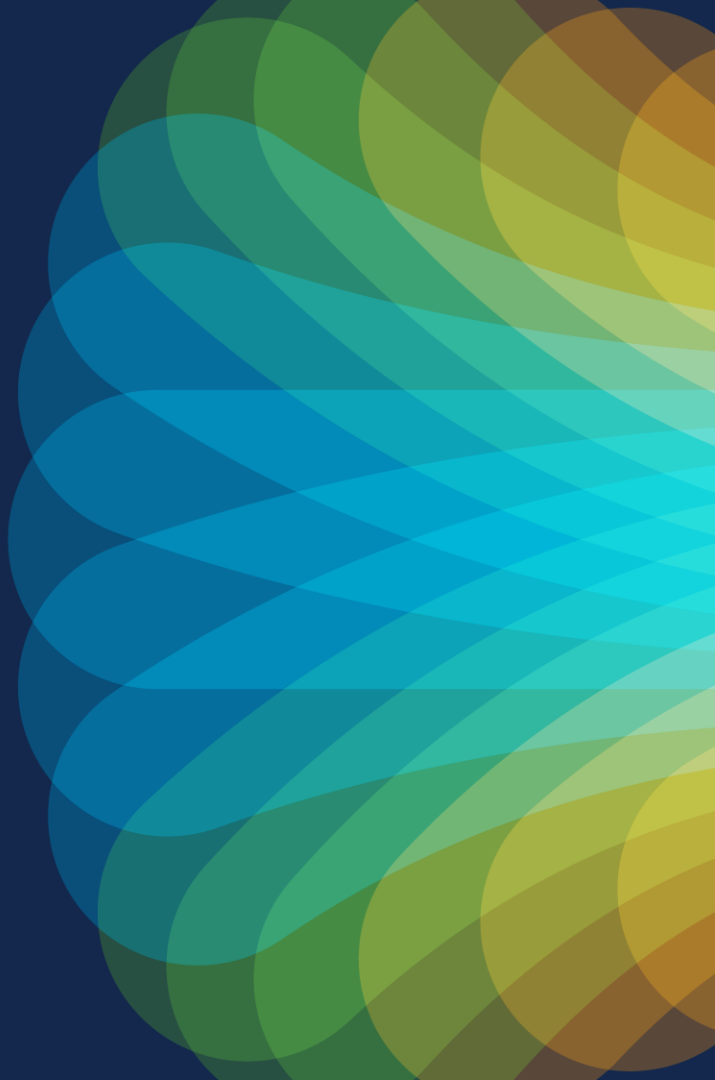
4. Enter into the container
    1. docker exec -it CONTAINER_ID /bin/bash
        1. Note: replace CONTAINER_ID with the ID found in step 3
5. Navigate to the correct folder
    1. cd
    2. cd cisco-ios-xe-panda-lab-terraform
6. Modify the device credentials in the header.tf file

```
sdeweese@SDEWEESE-M-C20V ~ % docker exec -it 288fa40fd38b /bin/bash
root@288fa40fd38b:/# cd
root@288fa40fd38b:~# ls
cisco-ios-xe-panda-lab-terraform  start.sh
root@288fa40fd38b:~# cd cisco-ios-xe-panda-lab-terraform
root@288fa40fd38b:~/cisco-ios-xe-panda-lab-terraform#
root@288fa40fd38b:~/cisco-ios-xe-panda-lab-terraform#
root@288fa40fd38b:~/cisco-ios-xe-panda-lab-terraform# ls
README.md  cpu.tf  header.tf  imgs  terraform.tfstate  terraform.tfstate.backup  terraform.tfvars  xpaths.tf
```

```
provider "iosxe" {
        username = "admin" ← replace with Cisco IOS XE device username
        password = "XXXXXXXX" ← replace with Cisco IOS XE device password
        url = "https://your-switch-hostname-or-ip" ← replace with Cisco IOS XE device hostname or I
}
```

7. Configure the Cisco IOS XE device using Terraform

Resources



dCloud Programmability

https://dcloud.cisco.com
"Cisco Catalyst 9000 IOS XE Programmability & Automation Lab v1"
https://dcloud2.cisco.com/demo/catalyst-9000-ios-xe-programmability-automation-lab-v1

Use Cases:

EVPN:

Ansible with CLI deployment of EVPN solutions
EVPN management over RESTCONF/YANG with Postman
Declarative EVPN fabric management with Terraform

Model Driven Telemetry

Telemetry configuration with CLI and YANG Suite
Collection with TIG_MDT container and tooling

YANG Programmability

YANG Suite tooling and integrations to YANG API's
Ansible integrations

Tooling and Integrations

YANG Suite
- NETCONF/RESTCONF/gNMI API
  - Ansible integration
- NETCONF/gNMI Dial-In Telemetry
- gRPC Dial-Out Telemetry receiver

Telemetry
- TIG stack in Docker
- Grafana dashboard for device health

Postman / RESTCONF
- EVPN fabric API calls

Terraform/RESTCONF
- Declarative EVPN fabric management

Ansible
- EVPN solution enablement using CLI

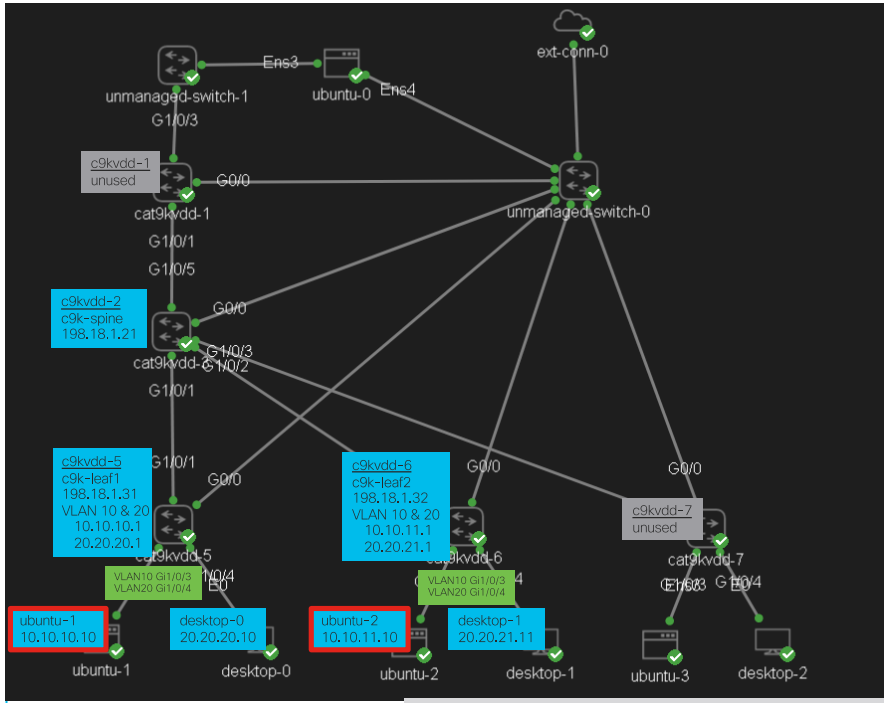
Ubuntu VM Details:

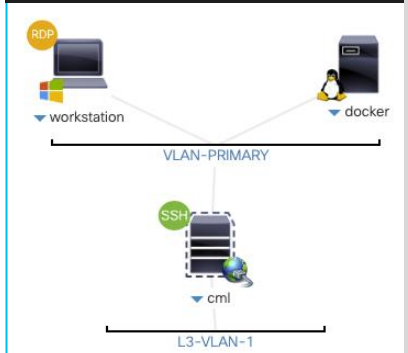
Syslog receiver from all switches
TFTP config backup
See slide

Windows VM Details

VS Code
Terraform @ folder
Ansible @ folder
Chrome browser
YANG Suite, Grafana
Bash/PS/Cmd shells
SSH into C9K or Ubuntu
Postman
Workspace for EVPN

3x C9K Virtual VM's





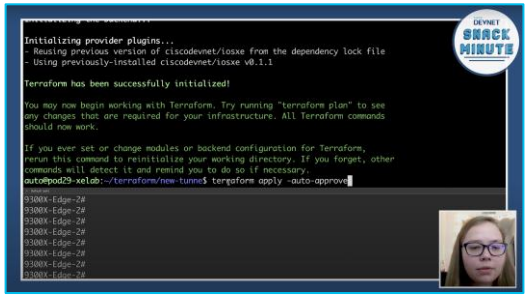
VLAN1
c9k-spine
IP: 198.18.1.21
developer / C1sco12345
c9k-leaf1
IP: 198.18.1.31
developer / C1sco12345
c9k-leaf2
IP: 198.18.1.32
developer / C1sco12345
c9kvdd-1 - unconfigured

c9kvdd-7 - unconfigured

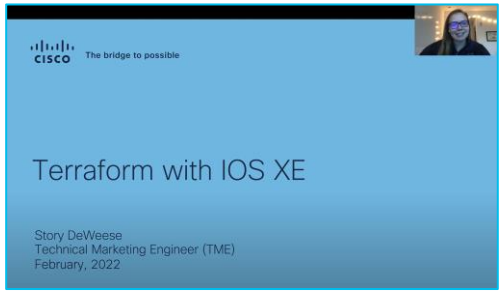
Blog and Resources: Terraform

https://github.com/CiscoDevNet/terraform-provider-iosxe/
https://registry.terraform.io/search/providers?namespace=CiscoDevNet

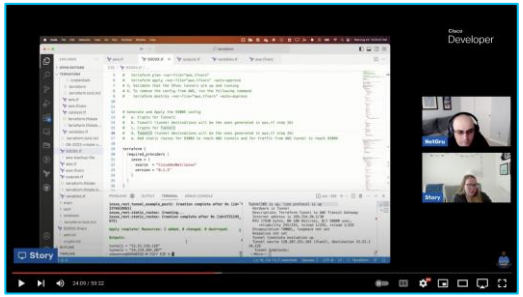
Questions? Join the Ask IOS XE Terraform Provider Webex space: https://eurl.io/#PtsT8eJFl



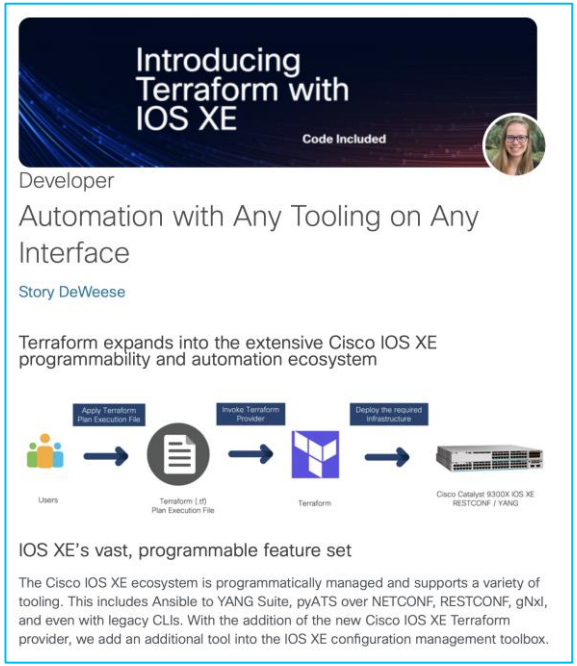
Demo Create a Crypto Tunnel Video:
https://www.youtube.com/watch?v=bPS0bhPacDw



Intro to IOS XE Terraform Provider Video:
https://www.youtube.com/watch?v=GEY_hyXimbA



Crypto IPsec Live Stream with DevRel:
https://www.youtube.com/watch?v=WkgDlE0HpXs



https://blogs.cisco.com/developer/terraformiosxe01

Cisco Catalyst IOS XE Programmability Sessions at Cisco Live Europe Amsterdam 2024

Monday Feb 5

11AM DEVLIT-2083: Automate Cisco IOS XE Device Configuration Using Terraform (Story)

1:30PM CISCOU-1024: Model Driven Telemetry with Cisco IOS XE made Simple (Jeremy)

3:30PM DEVLIT-1965: Automation using multiple API's in the IOS XE Device Programmability Lab (Jeremy)

Session Levels:
Beginner
Intermediate

Tuesday Feb 6

11:30AM DEVNET-1441: Catalyst 9000 Virtual topology simulation and configuration management solutions (Ama+Jeremy)

11:30AM DEVLIT-2787: Explore and visualize YANG models with YANG Suite (Story)

1:30PM DEVWKS-2270: Implementing Cisco IOS XE Model Driven Telemetry with Telegraf, InfluxDB, and Grafana (Jeremy)

1:30PM CISCOU-2013: Explore Cisco IOS XE Automation Tooling and Use Cases (Story)

4:30PM DEVLIT-2062: Getting Started with Secure Zero Touch Provisioning (Story)

Wednesday Feb 7

1:30PM DEVNET-2464: How to Become a Cisco IOS XE Terraform Expert (Story)

4:30PM CISCOU-2021: gNMI Oh My! Automation with the gRPC micro-services on IOS XE (Jeremy)

Thursday Feb 8

2:30PM BRKOPS-2455: Infrastructure as Code with Cisco Catalyst 9000 Virtual (Ama+Jeremy)

2:30PM DEVNET-1468: Programmability, Automation Model Driven Telemetry on Cisco IOS XE with a dash of YANG Suite (Story)

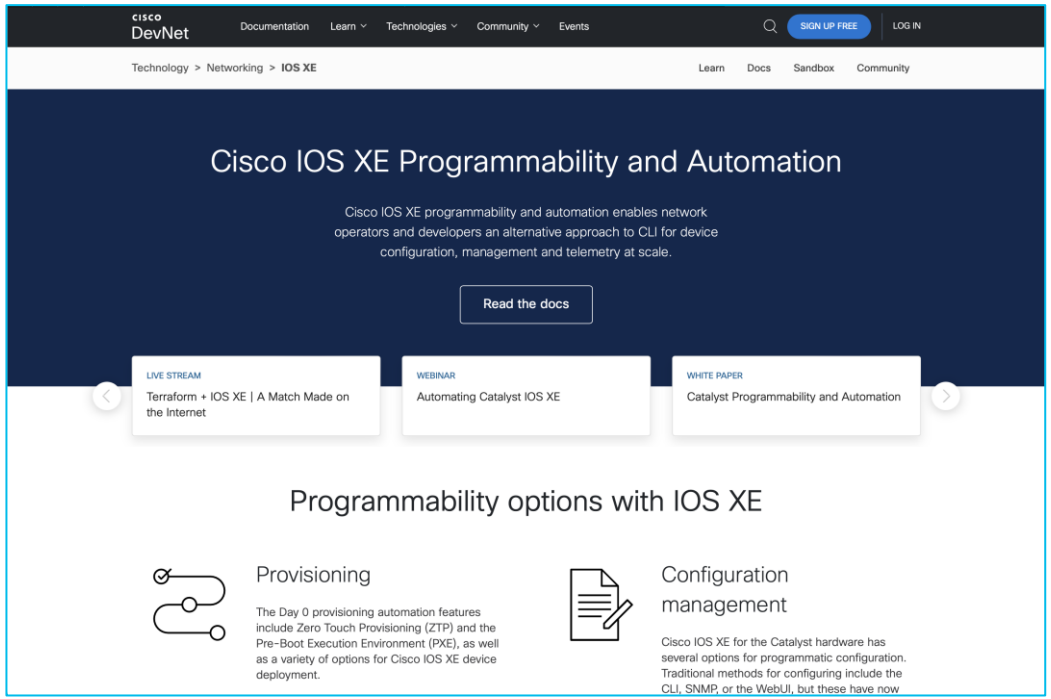
Friday Feb 9

Learn more about sessions this week and on-demand: https://blogs.cisco.com/developer/iosxeciscoliveemea2024

cisco Live!

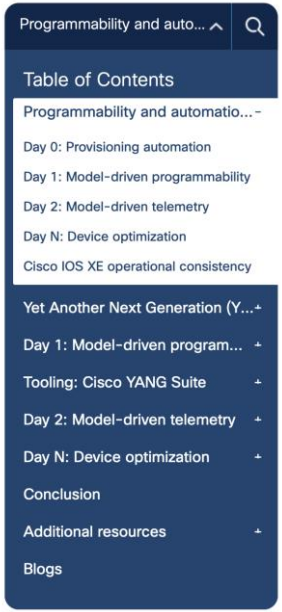
Cisco DevNet Site

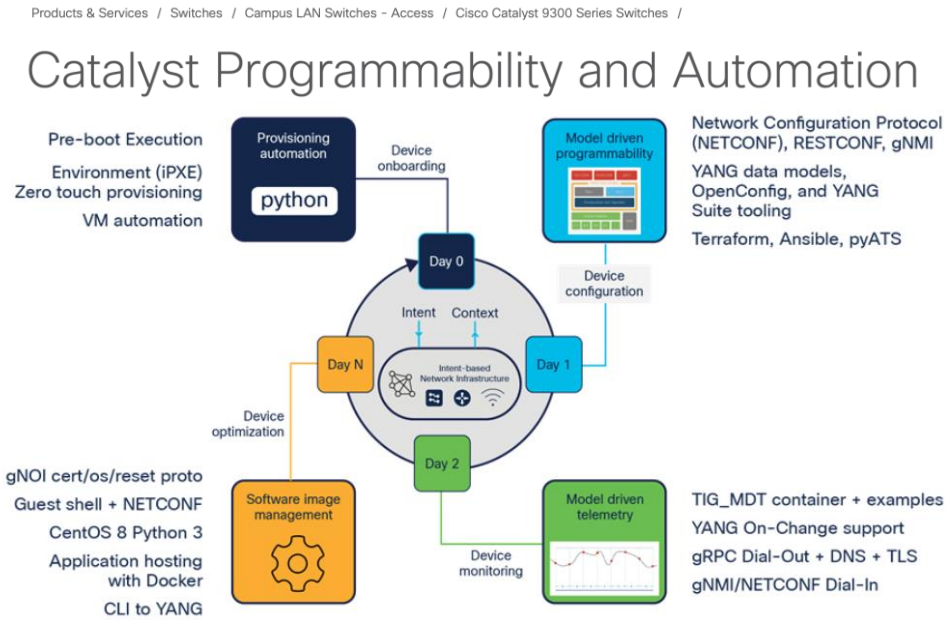
The one-stop-shop for Cisco IOS XE Programmability resources including videos, white papers, labs and more!



https://developer.cisco.com/iosxe/

White Paper – API





Website: https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-9300-series-switches/nb-06-catalyst-programmability-automation-wp.html
PDF: https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-9300-series-switches/nb-06-catalyst-programmability-automation-wp.pdf

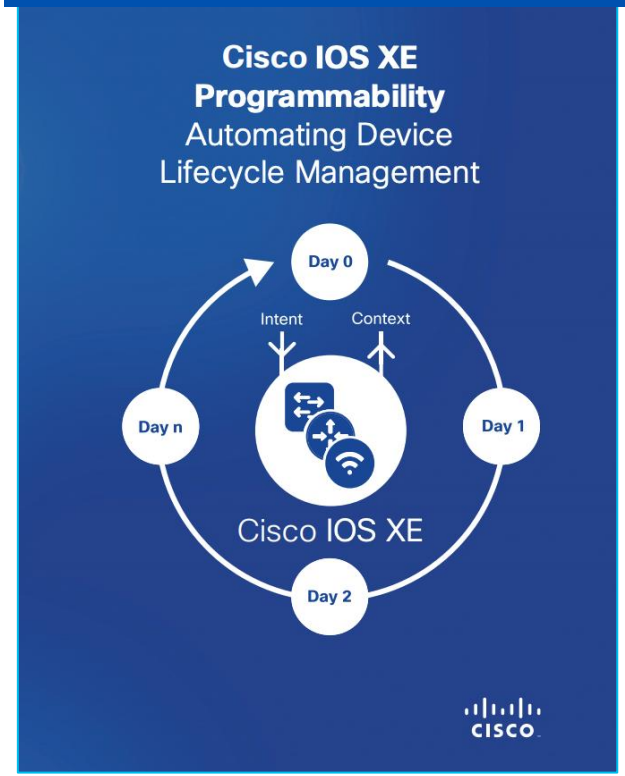
MDT White Paper coming soon!

http://cs.co/apiwp

http://cs.co/apiwppdf

Cisco IOS XE Programmability – Booksprint Book

http://cs.co/programmabilitybook OR https://www.cisco.com/c/dam/en/us/products/collateral/enterprise-networks/nb-06-ios-xe-prog-ebook-cte-en.pdf



Programmability Configuration Guide

Book Table of Contents

https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/prog/configuration/1713/b_1713_programmability_cg.html


cisco
The bridge to possible
Thank you
cisco Live!


cisco Live!
Let's go